Open Source Used In AppDynamics_ServoX 2.2.2

Cisco Systems, Inc.

www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-1603641232

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please submit this **form.**

In your requests please include the following reference number 78EE117C99-1603641232

Contents

1.1 tabulate 0.9.0

1.1.1 Available under license :

Copyright (c) 2011-2020 Sergey Astanin and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2 rich 13.3.2

1.2.1 Available under license :

Contributors

The following people have contributed to the development of Rich:

<!-- Add your name below, sort alphabetically by surname. Link to GitHub profile / your home page. -->

- [Patrick Arminio](https://github.com/patrick91)
- [Gregory Beauregard](https://github.com/GBeauregard/pyffstream)
- [Artur Borecki](https://github.com/pufereq)
- [Dennis Brakhane](https://github.com/brakhane)
- [Darren Burns](https://github.com/darrenburns)
- [Jim Crist-Harif](https://github.com/jcrist)
- [Ed Davis](https://github.com/davised)
- [Pete Davison](https://github.com/pd93)
- [James Estevez](https://github.com/jstvz)
- [Oleksis Fraga](https://github.com/oleksis)
- [Andy Gimblett](https://github.com/gimbo)
- [Micha Grny](https://github.com/mgorny)
- [Nok Lam Chan](https://github.com/noklam)
- [Leron Gray](https://github.com/daddycocoaman)
- [Andre Hora](https://github.com/andrehora)
- [Kenneth Hoste](https://github.com/boegel)
- [Lanqing Huang](https://github.com/lqhuang)
- [Finn Hughes](https://github.com/finnhughes)
- [Ionite](https://github.com/ionite34)
- [Josh Karpel](https://github.com/JoshKarpel)
- [Jan Katins](https://github.com/jankatins)
- [Hugo van Kemenade](https://github.com/hugovk)
- [Andrew Kettmann](https://github.com/akettmann)
- [Martin Larralde](https://github.com/althonos)
- [Hedy Li](https://github.com/hedythedev)
- [Luka Mamukashvili](https://github.com/UltraStudioLTD)
- [Alexander Mancevice](https://github.com/amancevice)
- [Will McGugan](https://github.com/willmcgugan)
- [Paul McGuire](https://github.com/ptmcg)

- [Antony Milne](https://github.com/AntonyMilneQB)
- [Michael Milton](https://github.com/multimeric)
- [Martina Oefelein](https://github.com/oefe)
- [Nathan Page](https://github.com/nathanrpage97)
- [Dave Pearson](https://github.com/davep/)
- [Avi Perl](https://github.com/avi-perl)
- [Laurent Peuch](https://github.com/psycojoker)
- [Ronny Pfannschmidt](https://github.com/RonnyPfannschmidt/)
- [Olivier Philippon](https://github.com/DrBenton)
- [Kylian Point](https://github.com/p0lux)
- [Kyle Pollina](https://github.com/kylepollina)
- [Sebastin Ramrez](https://github.com/tiangolo)
- [Felipe Guedes](https://github.com/guedesfelipe)
- [Min RK](https://github.com/minrk)
- [Clment Robert](https://github.com/neutrinoceros)
- [Brian Rutledge](https://github.com/bhrutledge)
- [Tushar Sadhwani](https://github.com/tusharsadhwani)
- [Paul Sanders](https://github.com/sanders41)
- [Tim Savage](https://github.com/timsavage)
- [Anthony Shaw](https://github.com/tonybaloney)
- [Nicolas Simonds](https://github.com/0xDEC0DE)
- [Aaron Stephens](https://github.com/aaronst)
- [Gabriele N. Tornetta](https://github.com/p403n1x87)
- [Nils Vu](https://github.com/nilsvu)
- [Arian Mollik Wasi](https://github.com/wasi-master)
- [Handhika Yanuar Pratama](https://github.com/theDreamer911)
- [za](https://github.com/za)
- [Motahhar Mokfi](https://github.com/motahhar)
- [Tomer Shalev](https://github.com/tomers)
- [Serkan UYSAL](https://github.com/uysalserkan)
- [Zhe Huang](https://github.com/onlyacat)
- [Adrian Zuber](https://github.com/xadrianzetx)
- [Ke Sun](https://github.com/ksun212)
- [Qiming Xu](https://github.com/xqm32)
- [James Addison](https://github.com/jayaddison)

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.3 urllib3 1.26.15

1.3.1 Available under license :

MIT License

Copyright (c) 2008-2020 Andrey Petrov and contributors (see CONTRIBUTORS.txt)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.4 chevron 0.14.0

1.4.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: chevron
Version: 0.14.0
Summary: Mustache templating language renderer
Home-page: https://github.com/noahmorrison/chevron
Author: noah morrison
Author-email: noah@morrison.ph
License: MIT
Description: [![PyPI version](https://badge.fury.io/py/chevron.svg)](https://badge.fury.io/py/chevron)
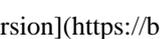
[![Build Status](https://travis-ci.org/noahmorrison/chevron.svg?branch=master)](https://travis-ci.org/noahmorrison/chevron)
[![Coverage Status](https://coveralls.io/repos/github/noahmorrison/chevron/badge.svg?branch=master)](https://coveralls.io/github/noahmorrison/chevron?branch=master)

A python implementation of the [mustache templating language](http://mustache.github.io).

Why chevron?

I'm glad you asked!

chevron is fast

Chevron runs in less than half the time of [pystache](http://github.com/defunkt/pystache) (Which is not even up to date on the spec).
And in about 70% the time of [Stache](https://github.com/hyperturtle/Stache) (A 'trimmed' version of mustache, also not spec compliant).

chevron is pep8

The flake8 command is run by [travis](https://travis-ci.org/noahmorrison/chevron) to ensure consistency.

chevron is spec compliant

Chevron passes all the unittests provided by the [spec](https://github.com/mustache/spec) (in every version listed below).

If you find a test that chevron does not pass, please [report it.](https://github.com/noahmorrison/chevron/issues/new)

chevron is Python 2 and 3 compatible

Python 2.6, 2.7, 3.2, 3.3, 3.4, 3.5, and 3.6 are all tested by travis.

USAGE

Commandline usage: (if installed via pypi)
```
usage: chevron [-h] [-v] [-d DATA] [-p PARTIALS_PATH] [-e PARTIALS_EXT]
               [-l DEF_LDEL] [-r DEF_RDEL]
               template

positional arguments:
```

```
  template          The mustache file

optional arguments:
 -h, --help          show this help message and exit
 -v, --version       show program's version number and exit
 -d DATA, --data DATA  The json data file
 -p PARTIALS_PATH, --path PARTIALS_PATH
              The directory where your partials reside
 -e PARTIALS_EXT, --ext PARTIALS_EXT
              The extension for your mustache partials, 'mustache'
              by default
 -l DEF_LDEL, --left-delimiter DEF_LDEL
              The default left delimiter, "{{" by default.
 -r DEF_RDEL, --right-delimiter DEF_RDEL
              The default right delimiter, "}}" by default.
```

Python usage with strings
```python
import chevron

chevron.render('Hello, {{ mustache }}!', {'mustache': 'World'})
```

Python usage with file
```python
import chevron

with open('file.mustache', 'r') as f:
    chevron.render(f, {'mustache': 'World'})
```

Python usage with unpacking
```python
import chevron

args = {
 'template': 'Hello, {{ mustache }}!',

 'data': {
   'mustache': 'World'
 }
}

chevron.render(**args)
```

chevron supports partials (via dictionaries)

```python
import chevron

args = {
    'template': 'Hello, {{> thing }}!',

    'partials_dict': {
        'thing': 'World'
    }
}

chevron.render(**args)
```

chevron supports partials (via the filesystem)
```python
import chevron

args = {
    'template': 'Hello, {{> thing }}!',

    # defaults to .
    'partials_path': 'partials/',

    # defaults to mustache
    'partials_ext': 'ms',
}

# ./partials/thing.ms will be read and rendered
chevron.render(**args)
```

chevron supports lambdas
```python
import chevron

def first(text, render):
    # return only first occurance of items
    result = render(text)
    return [ x.strip() for x in result.split(" || ") if x.strip() ][0]

def inject_x(text, render):
    # inject data into scope
    return render(text, {'x': 'data'})

args = {
    'template': 'Hello, {{# first}} {{x}} || {{y}} || {{z}} {{/ first}}!  {{# inject_x}} {{x}} {{/ inject_x}}',
```

```
        'data': {
            'y': 'foo',
            'z': 'bar',
            'first': first,
            'inject_x': inject_x
        }
    }

    chevron.render(**args)
```

INSTALL

- with git
```
$ git clone https://github.com/noahmorrison/chevron.git
```

or using submodules
```
$ git submodules add https://github.com/noahmorrison/chevron.git
```

Also available on pypi!

- with pip
```
$ pip install chevron
```

TODO
---

* get popular
* have people complain
* fix those complaints

Platform: UNKNOWN
Classifier: Development Status :: 4 - Beta
Classifier: License :: OSI Approved :: MIT License
Classifier: Programming Language :: Python :: 2.6
Classifier: Programming Language :: Python :: 2.7
Classifier: Programming Language :: Python :: 3.2
Classifier: Programming Language :: Python :: 3.3
Classifier: Programming Language :: Python :: 3.4

Classifier: Programming Language :: Python :: 3.5

Classifier: Programming Language :: Python :: 3.6

Classifier: Topic :: Text Processing :: Markup

Description-Content-Type: text/markdown

Found in path(s):

* /opt/cola/permits/1343905136_1655169490.754643/0/chevron-0-14-0-tar-gz/chevron-0.14.0/PKG-INFO

* /opt/cola/permits/1343905136_1655169490.754643/0/chevron-0-14-0-tar-gz/chevron-0.14.0/chevron.egg-info/PKG-INFO

No license file was found, but licenses were detected in source scan.

'License :: OSI Approved :: MIT License',

Found in path(s):

* /opt/cola/permits/1343905136_1655169490.754643/0/chevron-0-14-0-tar-gz/chevron-0.14.0/setup.py

1.5 mypy-extensions 1.0.0

1.5.1 Available under license :

Mypy extensions are licensed under the terms of the MIT license, reproduced below.

= = = = =

The MIT License

Copyright (c) 2016-2017 Jukka Lehtosalo and contributors

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

= = = = =

1.6 pydantic 1.10.6

1.6.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017, 2018, 2019, 2020, 2021 Samuel Colvin and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.7 python-stevedore 5.0.0

1.7.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.8 python-rsa 4.9

1.8.1 Available under license :

Copyright 2011 Sybren A. Stvel <sybren@stuvel.eu>

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.9 python-requests 2.28.2
1.9.1 Available under license :

Requests
Copyright 2019 Kenneth Reitz

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

1.10 types-toml 0.10.8.5

1.10.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1

Name: types-toml

Version: 0.10.8.5

Summary: Typing stubs for toml

Home-page: https://github.com/python/typeshed

License: Apache-2.0 license

Project-URL: GitHub, https://github.com/python/typeshed

Project-URL: Changes, https://github.com/typeshed-internal/stub_uploader/blob/main/data/changelogs/toml.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for toml

This is a PEP 561 type stub package for the `toml` package. It

can be used by type-checking tools like

[mypy](https://github.com/python/mypy/),

[pyright](https://github.com/microsoft/pyright),

[pytype](https://github.com/google/pytype/),

PyCharm, etc. to check code that uses

`toml`. The source for this package can be found at

https://github.com/python/typeshed/tree/main/stubs/toml. All fixes for

types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.

This package was generated from typeshed commit `880c0da4045cd5ff2c29b73429629adf27e49d50`.

Found in path(s):

* /opt/cola/permits/1603644312_1679389304.6939523/0/types-toml-0-10-8-5-tar-gz/types-toml-0.10.8.5/PKG-INFO

* /opt/cola/permits/1603644312_1679389304.6939523/0/types-toml-0-10-8-5-tar-gz/types-toml-

0.10.8.5/types_toml.egg-info/PKG-INFO

1.11 pytest 7.2.2

1.11.1 Available under license :

The MIT License (MIT)

Copyright (c) 2004 Holger Krekel and others

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.12 flake8-quotes 3.3.2

1.12.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.13 python-cffi 1.15.1

1.13.1 Available under license :

Except when otherwise stated (look for LICENSE files in directories or
information at the beginning of each file) all software and
documentation is licensed as follows:

The MIT License

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

1.14 types-aiofiles 23.1.0.0

1.14.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: types-aiofiles
Version: 23.1.0.0
Summary: Typing stubs for aiofiles
Home-page: https://github.com/python/typeshed
License: Apache-2.0 license
Project-URL: GitHub, https://github.com/python/typeshed
Project-URL: Changes, https://github.com/typeshed-internal/stub_uploader/blob/main/data/changelogs/aiofiles.md
Project-URL: Issue tracker, https://github.com/python/typeshed/issues
Project-URL: Chat, https://gitter.im/python/typing
Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for aiofiles

This is a PEP 561 type stub package for the `aiofiles` package. It

can be used by type-checking tools like

[mypy](https://github.com/python/mypy/),

[pyright](https://github.com/microsoft/pyright),

[pytype](https://github.com/google/pytype/),

PyCharm, etc. to check code that uses

`aiofiles`. The source for this package can be found at

https://github.com/python/typeshed/tree/main/stubs/aiofiles. All fixes for

types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.

This package was generated from typeshed commit `390058eb39414e3c53e1f523eec0f5889b5f1c84`.

Found in path(s):

* /opt/cola/permits/1603644479_1679525508.1857114/0/types-aiofiles-23-1-0-0-tar-gz/types-aiofiles-23.1.0.0/PKG-INFO

* /opt/cola/permits/1603644479_1679525508.1857114/0/types-aiofiles-23-1-0-0-tar-gz/types-aiofiles-23.1.0.0/types_aiofiles.egg-info/PKG-INFO

1.15 kubernetes-asyncio 22.6.5

1.15.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2014 The Kubernetes Authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.16 pyaml 21.10.1

1.16.1 Available under license :

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004

Copyright (C) 2012 Mike Kazantsev <mk.fraggod@gmail.com>

Everyone is permitted to copy and distribute verbatim or modified
copies of this license document, and changing it is allowed as long
as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

1.17 mdurl 0.1.2

1.17.1 Available under license :

Copyright (c) 2015 Vitaly Puzrin, Alex Kocharin.
Copyright (c) 2021 Taneli Hukkinen

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

.parse() is based on Joyent's node.js `url` code:

Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.18 python-distlib 0.3.6

1.18.1 Available under license :

A. HISTORY OF THE SOFTWARE

==========================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations (now Zope
Corporation, see http://www.zope.com). In 2001, the Python Software
Foundation (PSF, see http://www.python.org/psf/) was formed, a
non-profit organization created specifically to own Python-related
Intellectual Property. Zope Corporation is a sponsoring member of
the PSF.

All Python releases are Open Source (see http://www.opensource.org for
the Open Source Definition). Historically, most, but not all, Python
releases have also been GPL-compatible; the table below summarizes
the various releases.

Release	Derived from	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes

1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes
1.6	1.5.2	2000	CNRI	no
2.0	1.6	2000	BeOpen.com	no
1.6.1	1.6	2001	CNRI	yes (2)
2.1	2.0+1.6.1	2001	PSF	no
2.0.1	2.0+1.6.1	2001	PSF	yes
2.1.1	2.1+2.0.1	2001	PSF	yes
2.2	2.1.1	2001	PSF	yes
2.1.2	2.1.1	2002	PSF	yes
2.1.3	2.1.2	2002	PSF	yes
2.2.1	2.2	2002	PSF	yes
2.2.2	2.2.1	2002	PSF	yes
2.2.3	2.2.2	2003	PSF	yes
2.3	2.2.2	2002-2003	PSF	yes
2.3.1	2.3	2002-2003	PSF	yes
2.3.2	2.3.1	2002-2003	PSF	yes
2.3.3	2.3.2	2002-2003	PSF	yes
2.3.4	2.3.3	2004	PSF	yes
2.3.5	2.3.4	2005	PSF	yes
2.4	2.3	2004	PSF	yes
2.4.1	2.4	2005	PSF	yes
2.4.2	2.4.1	2005	PSF	yes
2.4.3	2.4.2	2006	PSF	yes
2.4.4	2.4.3	2006	PSF	yes
2.5	2.4	2006	PSF	yes
2.5.1	2.5	2007	PSF	yes
2.5.2	2.5.1	2008	PSF	yes
2.5.3	2.5.2	2008	PSF	yes
2.6	2.5	2008	PSF	yes
2.6.1	2.6	2008	PSF	yes
2.6.2	2.6.1	2009	PSF	yes
2.6.3	2.6.2	2009	PSF	yes
2.6.4	2.6.3	2009	PSF	yes
2.6.5	2.6.4	2010	PSF	yes
3.0	2.6	2008	PSF	yes
3.0.1	3.0	2009	PSF	yes
3.1	3.0.1	2009	PSF	yes
3.1.1	3.1	2009	PSF	yes
3.1.2	3.1	2010	PSF	yes
3.2	3.1	2010	PSF	yes

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under
the GPL. All Python licenses, unlike the GPL, let you distribute
a modified version without making your changes open source. The
GPL-compatible licenses make it possible to combine Python with
other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible,
   because its license has a choice of law clause. According to
   CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1
   is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's
direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
===

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010
Python Software Foundation; All Rights Reserved" are retained in Python alone or
in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National
Research Initiatives, having an office at 1895 Preston White Drive,
Reston, VA 20191 ("CNRI"), and the Individual or Organization
("Licensee") accessing and otherwise using Python 1.6.1 software in
source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a nonexclusive, royalty-free, world-wide
license to reproduce, analyze, test, perform and/or display publicly,
prepare derivative works, distribute, and otherwise use Python 1.6.1
alone or in any derivative version, provided, however, that CNRI's
License Agreement and CNRI's notice of copyright, i.e., "Copyright (c)
1995-2001 Corporation for National Research Initiatives; All Rights
Reserved" are retained in Python 1.6.1 alone or in any derivative
version prepared by Licensee. Alternately, in lieu of CNRI's License
Agreement, Licensee may substitute the following text (omitting the
quotes): "Python 1.6.1 is made available subject to the terms and
conditions in CNRI's License Agreement. This Agreement together with
Python 1.6.1 may be located on the Internet using the following
unique, persistent identifier (known as a handle): 1895.22/1013. This
Agreement may also be obtained from a proxy server on the Internet
using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python 1.6.1 or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

    ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that

both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Stichting Mathematisch
Centrum or CWI not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
=======================
Distutils2 Contributors
=======================

The Distutils2 project was started by Tarek Ziad and is currently
maintained by ric Araujo. Many people have contributed to the project.

distlib has started off using some of the code from distutil2.

If you're making a patch, please add your name below in alphabetical order,
and welcome into the Fellowship of the Packaging!

Thanks to:

- Rajiv Abraham
- Ali Afshar
- David Barnett
- Pior Bastida
- Anthony Baxter
- Erik Bray
- C. Titus Brown
- Francisco Martn Brugu
- Nicolas Cadou
- Godefroid Chapelle
- Julien Courteau
- Christophe Combelles
- Jason R. Coombs
- Pierre-Yves David
- Ned Deily
- Konrad Delong
- Josip Djolonga
- John Edmonds
- Andr Espaze
- Boris Feld
- Andrew Francis

- Hallvard B Furuseth
- Patrice Gauthier
- Yannick Gingras
- Filip Gruszczyski
- Walker Hale IV
- Alexandre Hamelin
- Kelsey Hightower
- Thomas Holmes
- Preston Holmes
- Christian Hudon
- Julien Jehannet
- Jeremy Kloth
- Thomas Kluyver
- Amos Latteier
- Mathieu Leduc-Hamel
- Pierre Paul Lefebvre
- Tshepang Lekhonkhobe
- Alain Leufroy
- Janusz Lewandowski
- Martin von Lwis
- Hugo Lopes Tavares
- Guillermo Lpez-Anglada
- Justin Love
- Simon Mathieu
- Carl Meyer
- Alexis Mtaireau
- Julien Miotte
- Zubin Mithra
- Derek McTavish Mounce
- Paul Moore
- Michael Mulich
- Louis Munro
- Gal Pasgrimaud
- George Peristerakis
- Mathieu Perreault
- Guillaume Pratte
- Sean Reifschneider
- Antoine Reversat
- Arc Riley
- C. Anthony Risinger
- Elson Rodriguez
- Luis Rojas
- Erik Rose
- Brian Rosner
- Vinay Sajip
- Victor Stinner
- Alexandre Vassalotti
- Nadeem Vawda

Copyright (C) 2013 by Test User.

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Vinay Sajip
not be used in advertising or publicity pertaining to distribution
of the software without specific, written prior permission.

VINAY SAJIP DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL
VINAY SAJIP BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR
ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN
AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
BSD-licensed.

1.19 charset-normalizer 3.1.0

1.19.1 Available under license :

Included and Redistributed Files
----

17 files are included in the source distribution tar. They are used to verify the standard functions of
this library. They are mandatory to run `pytest` but not required to make the lib usable after install.
They DO NOT guarantee that the detection-coverage will not regress.

Those are EITHER pulled from Wikipedia _(CC-BY-SA)_ OR public domain archive.
You SHALL NOT modify any of those files without explicit approval.
MIT License

Copyright (c) 2019 TAHRI Ahmed R.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.20 aiohttp 3.8.4

1.20.1 Available under license :

Copyright aio-libs contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
- Contributors -

A. Jesse Jiryu Davis
Adam Bannister
Adam Cooper
Adam Horacek
Adam Mills
Adrian Krupa
Adrin Chaves
Ahmed Tahri
Alan Tse
Alec Hanefeld
Alejandro Gmez
Aleksandr Danshyn
Aleksey Kutepov
Alex Hayes
Alex Key
Alex Khomchenko
Alex Kuzmenko
Alex Lisovoy
Alexander Bayandin
Alexander Karpinsky
Alexander Koshevoy
Alexander Malev
Alexander Mohr
Alexander Shorin

Alexander Travov

Alexandru Mihai

Alexey Firsov

Alexey Popravka

Alexey Stepanov

Amin Etesamian

Amit Tulshyan

Amy Boyle

Anas El Amraoui

Anders Melchiorsen

Andrei Ursulenko

Andrej Antonov

Andrew Leech

Andrew Lytvyn

Andrew Svetlov

Andrew Zhou

Andrii Soldatenko

Anes Abismail

Antoine Pietri

Anton Kasyanov

Anton Zhdan-Pushkin

Arseny Timoniq

Artem Yushkovskiy

Arthur Darcet

Austin Scola

Ben Bader

Ben Greiner

Ben Timby

Benedikt Reinartz

Bob Haddleton

Boris Feld

Boyi Chen

Brett Cannon

Brian Bouterse

Brian C. Lane

Brian Muller

Bruce Merry

Bruno Souza Cabral

Bryan Kok

Bryce Drennan

Carl George

Cecile Tonglet

Chien-Wei Huang

Chih-Yuan Chen

Chris AtLee

Chris Laws

Chris Moore

Christopher Schmitt

Claudiu Popa

Colin Dunklau

Cong Xu

Damien Nad

Dan King

Dan Xu

Daniel Garca

Daniel Grossmann-Kavanagh

Daniel Nelson

Danny Song

David Bibb

David Michael Brown

Denilson Amorim

Denis Matiychuk

Denis Moshensky

Dennis Kliban

Dima Veselov

Dimitar Dimitrov

Dmitriy Safonov

Dmitry Doroshev

Dmitry Erlikh

Dmitry Lukashin

Dmitry Marakasov

Dmitry Shamov

Dmitry Trofimov

Dmytro Bohomiakov

Dmytro Kuznetsov

Dustin J. Mitchell

Eduard Iskandarov

Eli Ribble

Elizabeth Leddy

Enrique Saez

Eric Sheng

Erich Healy

Erik Peterson

Eugene Chernyshov

Eugene Naydenov

Eugene Nikolaiev

Eugene Tolmachev

Evan Kepner

Evert Lammerts

Felix Yan

Fernanda Guimares

FichteFoll

Florian Scheffler

Franek Magiera

Frederik Gladhorn

Frederik Peter Aalund

Gabriel Tremblay

Gary Wilson Jr.

Gennady Andreyev

Georges Dubus

Greg Holt

Gregory Haynes

Gus Goulart

Gustavo Carneiro

Gnther Jena

Hans Adema

Harmon Y.

Harry Liu

Hiroshi Ogawa

Hrishikesh Paranjape

Hu Bo

Hugh Young

Hugo Herter

Hugo van Kemenade

Hynek Schlawack

Igor Alexandrov

Igor Davydenko

Igor Mozharovsky

Igor Pavlov

Illia Volochii

Ilya Chichak

Ilya Gruzinov

Ingmar Steen

Jacob Champion

Jaesung Lee

Jake Davis

Jakob Ackermann

Jakub Wilk

Jan Buchar

Jashandeep Sohi

Jens Steinhauser

Jeonghun Lee

Jeongkyu Shin

Jeroen van der Heijden

Jesus Cea

Jian Zeng

Jinkyu Yi

Joel Watts

Jon Nabozny

Jonas Krger Svensson

Jonas Obrist

Jonathan Wright

Jonny Tan

Joongi Kim

Jordan Borean

Josep Cugat

Josh Junon

Joshu Coats

Julia Tsemusheva

Julien Duponchelle

Jungkook Park

Junjie Tao

Junyeong Jeong

Justas Trimailovas

Justin Foo

Justin Turner Arthur

Kay Zheng

Kevin Samuel

Kimmo Parviainen-Jalanko

Kirill Klenov

Kirill Malovitsa

Konstantin Valetov

Krzysztof Blazewicz

Kyrylo Perevozchikov

Kyungmin Lee

Lars P. Sndergaard

Liu Hua

Louis-Philippe Huberdeau

Loc Lajeanne

Lu Gong

Lubomir Gelo

Ludovic Gasc

Luis Pedrosa

Lukasz Marcin Dobrzanski

Makc Belousow

Manuel Miranda

Marat Sharafutdinov

Marco Paolini

Mariano Anaya

Mariusz Masztalerczuk

Marko Kohtala

Martijn Pieters

Martin Melka

Martin Richard

Mathias Frjdman

Mathieu Dugr

Matthieu Hauglustaine

Matthieu Rigal

Meet Mangukiya

Michael Ihnatenko

Micha Grny

Mikhail Burshteyn

Mikhail Kashkin

Mikhail Lukyanchenko

Mikhail Nacharov

Misha Behersky

Mitchell Ferree

Morgan Delahaye-Prat

Moss Collum

Mun Gwan-gyeong

Navid Sheikhol

Nicolas Braem

Nikolay Kim

Nikolay Novik

Nikolay Tiunov

Nndor Mtravlgyi

Oisin Aylward

Olaf Conradi

Pahaz Blinov

Panagiotis Kolokotronis

Pankaj Pandey

Pau Freixes

Paul Colomiets

Paulius ileikis

Paulus Schoutsen

Pavel Kamaev

Pavel Polyakov

Pavel Sapezhko

Pavol Vargovk

Pawel Kowalski

Pawel Miech

Pepe Osca

Philipp A.

Pieter van Beek

Qiao Han

Rafael Viotti

Raphael Bialon

Ral Cumplido

Required Field

Robert Lu

Robert Nikolich

Roman Podoliaka

Samuel Colvin

Sean Hunt

Sebastian Acuna

Sebastian Hanula

Sebastian Hther

Sebastien Geffroy

SeongSoo Cho

Sergey Ninua

Sergey Skripnick

Serhii Charykov

Serhii Kostel

Serhiy Storchaka

Simon Kennedy

Sin-Woo Bang

Stanislas Plum

Stanislav Prokop

Stefan Tjarks

Stepan Pletnev

Stephan Jaensch

Stephen Cirelli

Stephen Granade

Steven Seguin

Sunghyun Hwang

Sunit Deshpande

Sviatoslav Bulbakha

Sviatoslav Sydorenko

Taha Jahangir

Taras Voinarovskyi

Terence Honles

Thanos Lefteris

Thijs Vermeir

Thomas Forbes

Thomas Grainger

Tolga Tezel

Tomasz Trebski

Toshiaki Tanaka

Trinh Hoang Nhu

Vadim Suharnikov

Vaibhav Sagar

Vamsi Krishna Avula

Vasiliy Faronov

Vasyl Baran

Viacheslav Greshilov

Victor Collod

Victor Kovtun

Vikas Kawadia

Viktor Danyliuk

Ville Skytt

Vincent Maillol

Vitalik Verhovodov

Vitaly Haritonsky

Vitaly Magerya

Vladimir Kamarzin

Vladimir Kozlovski

Vladimir Rutsky

Vladimir Shulyak

Vladimir Zakharov

Vladyslav Bohaichuk

Vladyslav Bondar

W. Trevor King

Wei Lin

Weiwei Wang

Will Fatherley

Will McGugan

Willem de Groot

William Grzybowski

William S.

Wilson Ong

Yang Zhou

Yannick Koechlin

Yannick Proux

Ye Cao

Yegor Roganov

Yifei Kong

Young-Ho Cha

Yuriy Shatrov

Yury Pliner

Yury Selivanov

Yusuke Tsutsumi

Yuval Ofir

Zeal Wierslee

Zlatan Sianica

1.21 pygments-python-syntax-highlighter 2.14.0

1.21.1 Available under license :

"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

'# New ports collection makefile for: python25\n' Comment

'# Date created: 3 July 2003\n' Comment

'# Whom: Hye-Shik Chang <perky@FreeBSD.org>\n' Comment

'#\n'   Comment

'# $FreeBSD: ports/lang/python25/Makefile,v 1.145 2007/10/03 23:22:04 edwin Exp $\n' Comment

'\n'   Text.Whitespace

'PORTNAME'   Name.Variable
'='   Operator
'\t'   Text.Whitespace
'python25'   Text
'\n'   Text.Whitespace

'PORTVERSION' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'2'   Literal.Number
'.5.1'   Text
'\n'   Text.Whitespace

'CATEGORIES' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'lang'   Text
' '   Text.Whitespace
'python'   Text
' '   Text.Whitespace
'ipv6'   Text
'\n'   Text.Whitespace

'MASTER_SITES' Name.Variable
'='   Operator
'\t'   Text.Whitespace

'${'   Literal.String.Interpol
'PYTHON_MASTER_SITES' Name.Variable
'}'   Literal.String.Interpol
'\n'   Text.Whitespace

'MASTER_SITE_SUBDIR' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_MASTER_SITE_SUBDIR' Name.Variable
'}'   Literal.String.Interpol
'\n'   Text.Whitespace

'DISTFILES'   Name.Variable
'='   Operator
'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_DISTFILE' Name.Variable
'}'   Literal.String.Interpol
'\n'   Text.Whitespace

'\n'   Text.Whitespace

'MAINTAINER' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'python@FreeBSD.org' Text
'\n'   Text.Whitespace

'COMMENT'   Name.Variable
'?='   Operator
'\t'   Text.Whitespace
'An'   Text
' '   Text.Whitespace
'interpreted' Text
' '   Text.Whitespace
'object-oriented' Text
' '   Text.Whitespace
'programming' Text
' '   Text.Whitespace
'language'   Text
'\n'   Text.Whitespace

'\n'   Text.Whitespace

'DIST_SUBDIR' Name.Variable
'='   Operator
'\t'   Text.Whitespace

```
'python'    Text
'\n'        Text.Whitespace

'WRKSRC'    Name.Variable
'='         Operator
'\t\t'      Text.Whitespace
'${'        Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'         Literal.String.Interpol
'/portbld.static' Text
'\n'        Text.Whitespace

'PATCH_WRKSRC' Name.Variable
'='         Operator
'\t'        Text.Whitespace
'${'        Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'         Literal.String.Interpol
'\n'        Text.Whitespace

'GNU_CONFIGURE' Name.Variable
'='         Operator
'\t'        Text.Whitespace
'yes'       Text
'\n'        Text.Whitespace

'CONFIGURE_TARGET' Name.Variable
'='         Operator
'\t'        Text.Whitespace
'--build'   Text
'='         Operator
'${'        Literal.String.Interpol
'MACHINE_ARCH' Name.Variable
'}'         Literal.String.Interpol
'-portbld-freebsd' Text
'${'        Literal.String.Interpol
'OSREL'     Name.Variable
'}'         Literal.String.Interpol
'\n'        Text.Whitespace

'CONFIGURE_SCRIPT' Name.Variable
'='         Operator
'\t'        Text.Whitespace
'../configure' Text
' '         Text.Whitespace
'# must be relative\n' Comment.Single

'CONFIGURE_ENV' Name.Variable
```

'='   Operator
'\t'   Text.Whitespace
'OPT'   Name.Variable
'='   Operator
'"'   Literal.String.Double
'${'   Literal.String.Interpol
'CFLAGS'   Name.Variable
'}'   Literal.String.Interpol
'"'   Literal.String.Double
' '   Text.Whitespace
'SVNVERSION' Name.Variable
'='   Operator
'"echo freebsd"' Literal.String.Double
'\n'   Text.Whitespace

'MAKE_ENV'   Name.Variable
'='   Operator
'\t'   Text.Whitespace
'VPATH'   Name.Variable
'='   Operator
'"'   Literal.String.Double
'${'   Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'"'   Literal.String.Double
'\n'   Text.Whitespace

'INSTALLS_SHLIB' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'yes'   Text
'\n'   Text.Whitespace

'INSTALL_TARGET' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'altinstall' Text
'\n'   Text.Whitespace

'MAN1'   Name.Variable
'='   Operator
'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.1'   Text
'\n'   Text.Whitespace

'\n'   Text.Whitespace

'USE_PYTHON' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'yes'   Text
'\n'   Text.Whitespace

'PYTHON_VERSION' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'python2.5' Text
'\n'   Text.Whitespace

'PYTHON_NO_DEPENDS' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'yes'   Text
'\n'   Text.Whitespace

'\n'   Text.Whitespace

'SHARED_WRKSRC' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/portbld.shared' Text
'\n'   Text.Whitespace

'PLIST'   Name.Variable
'='   Operator
'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/PLIST'   Text
'\n'   Text.Whitespace

'PLIST_TEMPLATE' Name.Variable
'?='   Operator
'${'   Literal.String.Interpol
'PKGDIR'   Name.Variable
'}'   Literal.String.Interpol
'/pkg-plist' Text
'\n'   Text.Whitespace

```
'PLIST_SUB'  Name.Variable
'='          Operator
'\t'         Text.Whitespace
'PYVER'      Name.Variable
'='          Operator
'${'         Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
':'          Punctuation
'S'          Name.Variable
'/python//'  Punctuation
'}'          Literal.String.Interpol
' '          Text.Whitespace
'\\\n'       Literal.String.Escape

'\t\t'       Text.Whitespace
'PYVER_WITHPAT' Name.Variable
'='          Operator
'${'         Literal.String.Interpol
'PORTVERSION' Name.Variable
':'          Punctuation
'S'          Name.Variable
'/.c/c/'     Punctuation
'}'          Literal.String.Interpol
'\n'         Text.Whitespace

'DEMODIR'    Name.Variable
'='          Operator
'\t'         Text.Whitespace
'${'         Literal.String.Interpol
'PREFIX'     Name.Variable
'}'          Literal.String.Interpol
'/share/examples/' Text
'${'         Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'          Literal.String.Interpol
'\n'         Text.Whitespace

'TOOLSDIR'   Name.Variable
'='          Operator
'\t'         Text.Whitespace
'${'         Literal.String.Interpol
'PREFIX'     Name.Variable
'}'          Literal.String.Interpol
'/share/'    Text
'${'         Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'          Literal.String.Interpol
'\n'         Text.Whitespace
```

'\n'   Text.Whitespace

'BIN_SCRIPTS' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'idle'   Text
' '   Text.Whitespace
'pydoc'   Text
' '   Text.Whitespace
'python'   Text
' '   Text.Whitespace
'python-shared' Text
' '   Text.Whitespace
'smtpd.py'   Text
' '   Text.Whitespace
'python-config' Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'python-shared-config' Text
'\n'   Text.Whitespace

'BINLINKS_SUB' Name.Variable
'='   Operator
'\t'   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'s,smtpd,smtpd${PYTHON_VER},'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'s,(idle|pydoc|python-shared|python),\\1${PYTHON_VER},'" Literal.String.Single
'\n\n'   Text.Whitespace

'OPTIONS'   Name.Variable
'='   Operator
'\t'   Text.Whitespace
'THREADS'   Text
' '   Text.Whitespace
'"Enable thread support"' Literal.String.Double
' '   Text.Whitespace
'on'   Text
' '   Text.Whitespace

'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'HUGE_STACK_SIZE' Text
' '   Text.Whitespace
'"Use a larger thread stack"' Literal.String.Double
' '   Text.Whitespace
'off'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'UCS4'   Text
' '   Text.Whitespace
'"Use UCS4 for unicode support"' Literal.String.Double
' '   Text.Whitespace
'on'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'PYMALLOC' Text
' '   Text.Whitespace
'"Use python\'s internal malloc"' Literal.String.Double
' '   Text.Whitespace
'on'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'IPV6'   Text
' '   Text.Whitespace
'"Enable IPv6 support"' Literal.String.Double
' '   Text.Whitespace
'on'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'FPECTL'   Text
' '   Text.Whitespace
'"Enable floating point exception handling"' Literal.String.Double
' '   Text.Whitespace
'off'   Text
'\n\n'   Text.Whitespace

'.include <bsd.port.pre.mk>\n' Comment.Preproc

'\n'   Text.Whitespace

'.if ${PYTHON_VERSION} == ${PYTHON_DEFAULT_VERSION}\n' Comment.Preproc

'MLINKS'   Name.Variable
'='   Operator
'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.1'   Text
' '   Text.Whitespace
'python.1'   Text
'\n'   Text.Whitespace

'PLIST_SUB'   Name.Variable
'+='   Operator
'\t'   Text.Whitespace
'IF_DEFAULT' Name.Variable
'='   Operator
'""'   Literal.String.Double
'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'PLIST_SUB'   Name.Variable
'+='   Operator
'\t'   Text.Whitespace
'IF_DEFAULT' Name.Variable
'='   Operator
'"@comment "' Literal.String.Double
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'# workaround for a bug in base curses.h.\n' Comment

'CFLAGS'   Name.Variable
'+='   Operator
'\t'   Text.Whitespace
'-D__wchar_t' Text
'='   Operator
'wchar_t'   Text
'\n'   Text.Whitespace

'\n'   Text.Whitespace

'.if !defined(WITHOUT_THREADS)\n' Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+=' 		Operator
'\t' 		Text.Whitespace
'--with-threads' Text
'\n' 		Text.Whitespace

'CFLAGS' 	Name.Variable
'+=' 		Operator
'\t\t' 		Text.Whitespace
'${' 		Literal.String.Interpol
'PTHREAD_CFLAGS' Name.Variable
'}' 		Literal.String.Interpol
'\n' 		Text.Whitespace

'.if defined(WITHOUT_HUGE_STACK_SIZE)\n' Comment.Preproc

'CFLAGS' 	Name.Variable
'+=' 		Operator
'\t\t' 		Text.Whitespace
'-DTHREAD_STACK_SIZE' Text
'=' 		Operator
'0x20000' 	Text
'\n' 		Text.Whitespace

'.else\n' 	Comment.Preproc

'CFLAGS' 	Name.Variable
'+=' 		Operator
'\t\t' 		Text.Whitespace
'-DTHREAD_STACK_SIZE' Text
'=' 		Operator
'0x100000' 	Text
'\n' 		Text.Whitespace

'.endif # defined(WITHOUT_HUGE_STACK_SIZE)\n' Comment.Preproc

'CONFIGURE_ENV' Name.Variable
'+=' 		Operator
'\t\t' 		Text.Whitespace
'LDFLAGS' 	Name.Variable
'=' 		Operator
'""' 		Literal.String.Double
'${' 		Literal.String.Interpol
'PTHREAD_LIBS' Name.Variable
'}' 		Literal.String.Interpol

```
' '          Literal.String.Double
'${'        Literal.String.Interpol
'LDFLAGS'    Name.Variable
'}'          Literal.String.Interpol
'"'          Literal.String.Double
'\n'         Text.Whitespace

'.else\n'    Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='        Operator
'\t'         Text.Whitespace
'--without-threads' Text
'\n'         Text.Whitespace

'.if defined(LDFLAGS)\n' Comment.Preproc

'CONFIGURE_ENV' Name.Variable
'+='        Operator
'\t\t'       Text.Whitespace
'LDFLAGS'    Name.Variable
'='          Operator
'"'          Literal.String.Double
'${'         Literal.String.Interpol
'LDFLAGS'    Name.Variable
'}'          Literal.String.Interpol
'"'          Literal.String.Double
'\n'         Text.Whitespace

'.endif # defined(LDFLAGS)\n' Comment.Preproc

'.endif # !defined(WITHOUT_THREADS)\n' Comment.Preproc

'\n'         Text.Whitespace

'.if !defined(WITHOUT_UCS4) && !defined(WITH_UCS2)\n' Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='        Operator
'\t'         Text.Whitespace
'--enable-unicode' Text
'='          Operator
'ucs4'       Text
'\n'         Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'         Text.Whitespace
```

'.if defined(WITHOUT_PYMALLOC)\n' Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='   Operator
'   ' Text.Whitespace
'--without-pymalloc' Text
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'.if ${ARCH} == i386\n' Comment.Preproc

'PLIST_SUB'   Name.Variable
'+='   Operator
'\t'   Text.Whitespace
'X86_ONLY'   Name.Variable
'='   Operator
'""'   Literal.String.Double
'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'PLIST_SUB'   Name.Variable
'+='   Operator
'\t'   Text.Whitespace
'X86_ONLY'   Name.Variable
'='   Operator
'"@comment "' Literal.String.Double
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'.if ${ARCH} == amd64 || ${ARCH} == ia64 || ${ARCH} == sparc64 || ${ARCH} == alpha\n' Comment.Preproc

'PLIST_SUB'   Name.Variable
'+='   Operator
'   ' Text.Whitespace
'32BIT_ONLY' Name.Variable
'='   Operator
'"@comment "' Literal.String.Double
'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'PLIST_SUB'   Name.Variable

'+='   Operator

' '   Text.Whitespace

'32BIT_ONLY' Name.Variable

'='   Operator

'""'   Literal.String.Double

'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'.if ${ARCH} == sparc64\n' Comment.Preproc

'CFLAGS'   Name.Variable

'+='   Operator

'   '   Text.Whitespace

'-DPYTHON_DEFAULT_RECURSION_LIMIT' Text

'='   Operator

'900'   Literal.Number

'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'.if !exists(/usr/bin/ypcat) # the world with NO_NIS\n' Comment.Preproc

'PLIST_SUB'   Name.Variable

'+='   Operator

'\t'   Text.Whitespace

'NO_NIS'   Name.Variable

'='   Operator

'"@comment "' Literal.String.Double

'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'PLIST_SUB'   Name.Variable

'+='   Operator

'\t'   Text.Whitespace

'NO_NIS'   Name.Variable

'='   Operator

'""'   Literal.String.Double

'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'.if !defined(WITHOUT_IPV6)\n' Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='   Operator
' '   Text.Whitespace
'--enable-ipv6' Text
'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='   Operator
' '   Text.Whitespace
'--disable-ipv6' Text
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'.if defined(WITH_FPECTL)\n' Comment.Preproc

'CONFIGURE_ARGS' Name.Variable
'+='   Operator
' '   Text.Whitespace
'--with-fpectl' Text
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'.if ${OSVERSION} >= 700000\n' Comment.Preproc

'PLATFORMS'   Name.Variable
'='   Operator
'plat-freebsd4' Text
' '   Text.Whitespace
'plat-freebsd5' Text
' '   Text.Whitespace
'plat-freebsd6' Text
'\n'   Text.Whitespace

'.elif ${OSVERSION} >= 600000\n' Comment.Preproc

'PLATFORMS'   Name.Variable
'='   Operator
'plat-freebsd4' Text
' '   Text.Whitespace

'plat-freebsd5' Text
' '   Text.Whitespace
'plat-freebsd7' Text
'\n'   Text.Whitespace

'.else\n'   Comment.Preproc

'PLATFORMS'   Name.Variable
'='   Operator
'plat-freebsd4' Text
' '   Text.Whitespace
'plat-freebsd6' Text
' '   Text.Whitespace
'plat-freebsd7' Text
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'pre-patch'   Name.Function
':'   Operator
'\n'   Text.Whitespace

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'MKDIR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Modules'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'SED'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'1s,^.*$$,#!${PREFIX}/bin/${PYTHON_VERSION},'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Tools/scripts/pydoc' Text
' '   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/pydoc2.5' Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'SED'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'1s,^.*$$,#!${PREFIX}/bin/${PYTHON_VERSION},'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Tools/scripts/idle' Text
' '   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/idle2.5' Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'SED'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'1s,^.*$$,#!${PREFIX}/bin/${PYTHON_VERSION},'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol

'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Lib/smtpd.py' Text
' '   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/smtpd2.5.py' Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'REINPLACE_CMD' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
"'s,/usr/doc/python-docs-,${PREFIX}/share/doc/python,g'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Lib/pydoc.py' Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'REINPLACE_CMD' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'\'s|^\\(*prefixes = .*\\)\\]$$|\\1, "${X11BASE}"]|g\"' Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Lib/site.py' Text

'\n\t'	Text.Whitespace
'${'	Literal.String.Interpol
'REINPLACE_CMD' Name.Variable
'}'	Literal.String.Interpol
' '	Text.Whitespace
'-e'	Text
' '	Text.Whitespace
'\\\n'	Literal.String.Escape

'\t\t'	Text.Whitespace
"'s|^\t\\(..ASDLGEN.*\\)$$|\t${TRUE}|g'" Literal.String.Single
' '	Text.Whitespace
'\\\n'	Literal.String.Escape

'\t\t'	Text.Whitespace
'${'	Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'	Literal.String.Interpol
'/Makefile.pre.in' Text
'\n\n\t'	Text.Whitespace
'${'	Literal.String.Interpol
'REINPLACE_CMD' Name.Variable
'}'	Literal.String.Interpol
' '	Text.Whitespace
'-e'	Text
' '	Text.Whitespace
'\\\n'	Literal.String.Escape

'\t\t'	Text.Whitespace
"'s|*\\(..INSTALL_SCRIPT.*\\)python-config$$|#port \\1|'" Literal.String.Single
' '	Text.Whitespace
'\\\n'	Literal.String.Escape

'\t\t'	Text.Whitespace
'${'	Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'	Literal.String.Interpol
'/Makefile.pre.in' Text
'\n\n\t'	Text.Whitespace
'${'	Literal.String.Interpol
'SED'	Name.Variable
'}'	Literal.String.Interpol
' '	Text.Whitespace
'-e'	Text
' '	Text.Whitespace
"'s|^#!.*|#!${PREFIX}/bin/${PYTHON_VERSION}|'" Literal.String.Single
' '	Text.Whitespace
'\\\n'	Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Misc/python-config.in' Text
' '   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'-config'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'SED'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-e'   Text
' '   Text.Whitespace
"'s|^#!.*|#!${PREFIX}/bin/${PYTHON_VERSION:S/thon/thon-shared/}|'" Literal.String.Single
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PATCH_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Misc/python-config.in' Text
' '   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
':'   Punctuation
'S'   Name.Variable
'/thon/thon-shared/' Punctuation
'}'   Literal.String.Interpol
'-config'   Text
'\n\n'   Text.Whitespace

'.if defined(WITH_FPECTL) && ${ARCH} == i386\n' Comment.Preproc

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'MKDIR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
'/Modules'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'ECHO'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'"fpectl fpectlmodule.c"' Literal.String.Double
' '   Text.Whitespace
'>>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
'/Modules/Setup.dist' Text
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n'   Text.Whitespace

'post-configure' Name.Function
':'   Operator
'\n'   Text.Whitespace

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'TAR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-C'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-cf'   Text
' '   Text.Whitespace

```
'-'         Text
' '         Text.Whitespace
'.'         Text
' '         Text.Whitespace
'|'         Punctuation
' '         Text.Whitespace
'${'        Literal.String.Interpol
'TAR'       Name.Variable
'}'         Literal.String.Interpol
' '         Text.Whitespace
'-C'        Text
' '         Text.Whitespace
'${'        Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'         Literal.String.Interpol
' '         Text.Whitespace
'-xf'       Text
' '         Text.Whitespace
'-'         Text
'\n\t'      Text.Whitespace
'${'        Literal.String.Interpol
'LN'        Name.Variable
'}'         Literal.String.Interpol
' '         Text.Whitespace
'-sf'       Text
' '         Text.Whitespace
'${'        Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'         Literal.String.Interpol
'/Lib'      Text
' '         Text.Whitespace
'${'        Literal.String.Interpol
'WRKSRC'    Name.Variable
'}'         Literal.String.Interpol
'/Lib'      Text
'\n\t'      Text.Whitespace
'${'        Literal.String.Interpol
'SED'       Name.Variable
'}'         Literal.String.Interpol
' '         Text.Whitespace
'-e'        Text
' '         Text.Whitespace
"'s,^\\(LDLIBRARY=\\).*$$,\\1libpython$$(VERSION).so,'" Literal.String.Single
' '         Text.Whitespace
'\\\n'      Literal.String.Escape

'\t\t'      Text.Whitespace
'-e'        Text
```

```
' '             Text.Whitespace
"'s,^\\(BLDLIBRARY=\\).*$$,\\1-L. -lpython$$(VERSION),'" Literal.String.Single
' '             Text.Whitespace
'\\\n'          Literal.String.Escape

'\t\t'          Text.Whitespace
'-e'            Text
' '             Text.Whitespace
"'s,^\\(CFLAGSFORSHARED=\\).*$$,\\1$$(CCSHARED),'" Literal.String.Single
' '             Text.Whitespace
'\\\n'          Literal.String.Escape

'\t\t'          Text.Whitespace
'-e'            Text
' '             Text.Whitespace
"'s,^\\(Makefile Modules/config.c:.*\\)Makefile.pre,\\1,'" Literal.String.Single
' '             Text.Whitespace
'\\\n'          Literal.String.Escape

'\t\t'          Text.Whitespace
'-e'            Text
' '             Text.Whitespace
"'s,^\\(.(BUILDPYTHON)\\: .*\\).(LIBRARY),\\1,'" Literal.String.Single
' '             Text.Whitespace
'\\\n'          Literal.String.Escape

'\t\t'          Text.Whitespace
'-e'            Text
' '             Text.Whitespace
"'s,^\\(.(BUILDPYTHON):.*\\).(LIBRARY),\\1,'" Literal.String.Single
' '             Text.Whitespace
'\\\n'          Literal.String.Escape

'\t\t'          Text.Whitespace
'${'            Literal.String.Interpol
'WRKSRC'        Name.Variable
'}'             Literal.String.Interpol
'/Makefile'     Text
' '             Text.Whitespace
'>'             Text
' '             Text.Whitespace
'${'            Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'             Literal.String.Interpol
'/Makefile'     Text
'\n\n'          Text.Whitespace

'pre-build'     Name.Function
```

```
':'        Operator
'\n'       Text.Whitespace

'\t'       Text.Whitespace
'cd'       Name.Builtin
' '        Text.Whitespace
'${'       Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'        Literal.String.Interpol
';'        Punctuation
' '        Text.Whitespace
'\\\n'     Literal.String.Escape

'\t'       Text.Whitespace
'${'       Literal.String.Interpol
'SETENV'   Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'${'       Literal.String.Interpol
'MAKE_ENV' Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'${'       Literal.String.Interpol
'MAKE'     Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'lib'      Text
'${'       Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'        Literal.String.Interpol
'.so'      Text
' '        Text.Whitespace
'python'   Text
';'        Punctuation
' '        Text.Whitespace
'\\\n'     Literal.String.Escape

'\t'       Text.Whitespace
'${'       Literal.String.Interpol
'LN'       Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'-f'       Text
' '        Text.Whitespace
'lib'      Text
'${'       Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'        Literal.String.Interpol
```

'.so'   Text
' '   Text.Whitespace
'lib'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.so.1'   Text
';'   Punctuation
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'LN'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-f'   Text
' '   Text.Whitespace
'python'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
':'   Punctuation
'S'   Name.Variable
'/thon/thon-shared/' Punctuation
'}'   Literal.String.Interpol
'\n\n'   Text.Whitespace

'pre-su-install' Name.Function
':'   Operator
'\n'   Text.Whitespace

'.for platform in ${PLATFORMS}\n' Comment.Preproc

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'MKDIR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHONPREFIX_LIBDIR' Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'platform' Name.Variable
'}'   Literal.String.Interpol
'\n'   Text.Whitespace

'.for file in IN.py regen\n' Comment.Preproc

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_DATA' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
'/Lib/'   Text
'${'   Literal.String.Interpol
'platform'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'file'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHONPREFIX_LIBDIR' Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'platform'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'\n'   Text.Whitespace

'.endfor\n' Comment.Preproc

'.endfor\n' Comment.Preproc

'\n'   Text.Whitespace

'pre-install' Name.Function
':'   Operator
'\n'   Text.Whitespace

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'CAT'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol

'PLIST_TEMPLATE' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'|'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'AWK'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'\'{ print $$0; } \\\n\t/LIBDIR.*\\.py$$/ && !/\/bad/ { print $$0 "o"; print $$0 "c"; }\"' Literal.String.Single
'\t'   Text.Whitespace
'>'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PLIST'   Name.Variable
'}'   Literal.String.Interpol
'\n\n'   Text.Whitespace

'\t@# if openssl 0.9.8 is detected, _sha{256,512} module won\'t be installed\n' Comment

'\t'   Text.Whitespace
'('   Operator
'['   Operator
' '   Text.Whitespace
'-f'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKSRC'   Name.Variable
'}'   Literal.String.Interpol
'/.without_own_sha' Text
' '   Text.Whitespace
']'   Operator
' '   Text.Whitespace
'&&'   Operator
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'GREP'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-v'   Text
' '   Text.Whitespace
"'lib-dynload/_sha'" Literal.String.Single
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PLIST'   Name.Variable

'}'   Literal.String.Interpol

' '   Text.Whitespace

'>'   Text

' '   Text.Whitespace

'${'   Literal.String.Interpol

'PLIST'   Name.Variable

'}'   Literal.String.Interpol

'.tmp'   Text

' '   Text.Whitespace

'&&'   Operator

' '   Text.Whitespace

'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace

'${'   Literal.String.Interpol

'CAT'   Name.Variable

'}'   Literal.String.Interpol

' '   Text.Whitespace

'${'   Literal.String.Interpol

'PLIST'   Name.Variable

'}'   Literal.String.Interpol

'.tmp'   Text

' '   Text.Whitespace

'>'   Text

' '   Text.Whitespace

'${'   Literal.String.Interpol

'PLIST'   Name.Variable

'}'   Literal.String.Interpol

')'   Operator

' '   Text.Whitespace

'||'   Operator

' '   Text.Whitespace

'${'   Literal.String.Interpol

'TRUE'   Name.Variable

'}'   Literal.String.Interpol

'\n\n'   Text.Whitespace

'post-install' Name.Function

':'   Operator

'\n'   Text.Whitespace

'\t@# install config providers\n' Comment

'\t'   Text.Whitespace

'${'   Literal.String.Interpol

'INSTALL_SCRIPT' Name.Variable

'}'   Literal.String.Interpol

' '   Text.Whitespace

'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'-config'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/bin'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_SCRIPT' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
':'   Punctuation
'S'   Name.Variable
'/thon/thon-shared/' Punctuation
'}'   Literal.String.Interpol
'-config'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/bin'   Text
'\n\n'   Text.Whitespace

'\t@# shared version of executable and library\n' Comment

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_PROGRAM' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/lib'   Text
'${'   Literal.String.Interpol

'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.so.1'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/lib'   Text
'\n\t'   Text.Whitespace
'cd'   Name.Builtin
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/lib'   Text
';'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'LN'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-sf'   Text
' '   Text.Whitespace
'lib'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.so.1'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'lib'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.so'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'LN'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-sf'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol

'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/lib/lib'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.so'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHONPREFIX_LIBDIR' Name.Variable
'}'   Literal.String.Interpol
'/config'   Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_PROGRAM' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'SHARED_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/'   Text
'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
':'   Punctuation
'S'   Name.Variable
'/thon/thon-shared/' Punctuation
'}'   Literal.String.Interpol
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/bin'   Text
'\n\n'   Text.Whitespace

'\t@# additional files installing by ports\n' Comment

'\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_SCRIPT' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol

'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/pydoc2.5' Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/idle2.5'   Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'WRKDIR'   Name.Variable
'}'   Literal.String.Interpol
'/smtpd2.5.py' Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PREFIX'   Name.Variable
'}'   Literal.String.Interpol
'/bin'   Text
'\n\t'   Text.Whitespace
'@'   Text
'${'   Literal.String.Interpol
'MKDIR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'MANPREFIX' Name.Variable
'}'   Literal.String.Interpol
'/man/man1' Text
'\n\t'   Text.Whitespace
'${'   Literal.String.Interpol
'INSTALL_MAN' Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Misc/python.man' Text
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'${'   Literal.String.Interpol
'MANPREFIX' Name.Variable
'}'   Literal.String.Interpol
'/man/man1/' Text

'${'   Literal.String.Interpol
'PYTHON_VERSION' Name.Variable
'}'   Literal.String.Interpol
'.1'   Text
'\n\n'   Text.Whitespace

'.if ${PYTHON_VERSION} == ${PYTHON_DEFAULT_VERSION}\n' Comment.Preproc

'\t'   Text.Whitespace
'for'   Keyword
' '   Text.Whitespace
'f'   Text
' '   Text.Whitespace
'in'   Keyword
' '   Text.Whitespace
'${'   Literal.String.Interpol
'BIN_SCRIPTS' Name.Variable
'}'   Literal.String.Interpol
';'   Punctuation
' '   Text.Whitespace
'do'   Keyword
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'TARGET'   Name.Variable
'='   Operator
'`'   Literal.String.Backtick
'${'   Literal.String.Interpol
'ECHO_CMD'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'$$'   Name.Variable
'f'   Text
' '   Text.Whitespace
'|'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'SED'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-E'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'BINLINKS_SUB' Name.Variable
'}'   Literal.String.Interpol
'`'   Literal.String.Backtick
';'   Punctuation

```
' '        Text.Whitespace
'\\\n'     Literal.String.Escape

'\t\t'     Text.Whitespace
'cd'       Name.Builtin
' '        Text.Whitespace
'${'       Literal.String.Interpol
'PREFIX'   Name.Variable
'}'        Literal.String.Interpol
'/bin'     Text
' '        Text.Whitespace
'&&'       Operator
' '        Text.Whitespace
'${'       Literal.String.Interpol
'LN'       Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'-f'       Text
' '        Text.Whitespace
'$$'       Name.Variable
'TARGET'   Text
' '        Text.Whitespace
'$$'       Name.Variable
'f'        Text
';'        Punctuation
' '        Text.Whitespace
'\\\n'     Literal.String.Escape

'\t'       Text.Whitespace
'done'     Keyword
'\n'       Text.Whitespace

'.endif\n'  Comment.Preproc

'\n'       Text.Whitespace

'.if !defined(NOPORTDOCS)\n' Comment.Preproc

'\t'       Text.Whitespace
'@'        Text
'${'       Literal.String.Interpol
'MKDIR'    Name.Variable
'}'        Literal.String.Interpol
' '        Text.Whitespace
'${'       Literal.String.Interpol
'TOOLSDIR'  Name.Variable
'}'        Literal.String.Interpol
'\n\t'     Text.Whitespace
```

'@cd'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
';'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'TAR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-cf'   Text
' '   Text.Whitespace
'-'   Text
' '   Text.Whitespace
'Tools'   Text
' '   Text.Whitespace
'|'   Punctuation
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'('   Operator
'cd'   Name.Builtin
' '   Text.Whitespace
'${'   Literal.String.Interpol
'TOOLSDIR'   Name.Variable
'}'   Literal.String.Interpol
';'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'TAR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-xf'   Text
' '   Text.Whitespace
'-'   Text
')'   Operator
'\n\t'   Text.Whitespace
'@'   Text
'${'   Literal.String.Interpol
'MKDIR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'${'   Literal.String.Interpol
'DEMODIR'   Name.Variable
'}'   Literal.String.Interpol
'\n\t'   Text.Whitespace

'@cd'   Text
' '   Text.Whitespace
'${'   Literal.String.Interpol
'PYTHON_WRKSRC' Name.Variable
'}'   Literal.String.Interpol
'/Demo'   Text
';'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'TAR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-cf'   Text
' '   Text.Whitespace
'-'   Text
' '   Text.Whitespace
'*'   Text
' '   Text.Whitespace
'|'   Punctuation
' '   Text.Whitespace
'\\\n'   Literal.String.Escape

'\t\t'   Text.Whitespace
'('   Operator
'cd'   Name.Builtin
' '   Text.Whitespace
'${'   Literal.String.Interpol
'DEMODIR'   Name.Variable
'}'   Literal.String.Interpol
';'   Punctuation
' '   Text.Whitespace
'${'   Literal.String.Interpol
'TAR'   Name.Variable
'}'   Literal.String.Interpol
' '   Text.Whitespace
'-xf'   Text
' '   Text.Whitespace
'-'   Text
')'   Operator
'\n'   Text.Whitespace

'.endif\n'   Comment.Preproc

'\n\t'   Text.Whitespace
'@'   Text
'${'   Literal.String.Interpol
'CAT'   Name.Variable
'}'   Literal.String.Interpol

```
' '         Text.Whitespace
'${'        Literal.String.Interpol
'PKGMESSAGE'  Name.Variable
'}'         Literal.String.Interpol
'\n\n'      Text.Whitespace

'.include <bsd.port.post.mk>\n' Comment.Preproc
```

1.22 cattrs 22.2.0

1.22.1 Available under license :

MIT License

Copyright (c) 2016, Tin Tvrtkovi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.23 tomlkit 0.11.6

1.23.1 Available under license :

Copyright (c) 2018 Sbastien Eustace

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

1.24 httpcore 0.16.3

1.24.1 Available under license :

1.25 flake8-pytest-style 1.7.2

1.25.1 Available under license :

```
********************************************************************************
https://github.com/m-burst/cookiecutter-pypackage-poetry
********************************************************************************
```

copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.26 anyio 3.6.2

1.26.1 Available under license :

The MIT License (MIT)

Copyright (c) 2018 Alex Grnholm

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.27 strenum 0.4.10

1.27.1 Available under license :

MIT License

Copyright (c) 2019 James C Sinclair

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.28 python-attrs 22.2.0

1.28.1 Available under license :

License and Credits

attrs is licensed under the [MIT](https://choosealicense.com/licenses/mit/) license.
The full license text can be also found in the [source code repository](https://github.com/python-attrs/attrs/blob/main/LICENSE).

attrs is written and maintained by [Hynek Schlawack](https://hynek.me/).

The development is kindly supported by my employer [Variomedia AG](https://www.variomedia.de/), *attrs*
[Tidelift subscribers](https://tidelift.com/subscription/pkg/pypi-attrs?utm_source=pypi-attrs&utm_medium=referral&utm_campaign=enterprise&utm_term=repo), and all my amazing [GitHub
Sponsors](https://github.com/sponsors/hynek).

A full list of contributors can be found in [GitHub's overview](https://github.com/python-attrs/attrs/graphs/contributors).

Its the spiritual successor of [characteristic](https://characteristic.readthedocs.io/) and aspires to fix some of it
clunkiness and unfortunate decisions.
Both were inspired by Twisteds
[FancyEqMixin](https://docs.twisted.org/en/stable/api/twisted.python.util.FancyEqMixin.html) but both are
implemented using class decorators because [subclassing is bad for
you](https://www.youtube.com/watch?v=3MNVP9-hglc), mkay?
The MIT License (MIT)

Copyright (c) 2015 Hynek Schlawack and the attrs contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.29 exceptiongroup 1.1.1
1.29.1 Available under license :

The MIT License (MIT)

Copyright (c) 2022 Alex Grnholm

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This project contains code copied from the Python standard library.
The following is the required license notice for those parts.

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and

its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022 Python Software Foundation;
All Rights Reserved" are retained in Python alone or in any derivative version
prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

1.30 aiofiles 23.1.0

1.30.1 Available under license :

Asyncio support for files

Copyright 2016 Tin Tvrtkovic

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{ }"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.31 uvicorn 0.21.1

1.31.1 Available under license :

Copyright 2017-present, [Encode OSS Ltd](https://www.encode.io/).
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
 contributors may be used to endorse or promote products derived from
 this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.32 asn-1-library-for-python 0.4.8

1.32.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 1.2

Name: pyasn1

Version: 0.4.8

Summary: ASN.1 types and codecs

Home-page: https://github.com/etingof/pyasn1

Author: Ilya Etingof

Author-email: etingof@gmail.com

Maintainer: Ilya Etingof <etingof@gmail.com>

License: BSD

Description: Pure-Python implementation of ASN.1 types and DER/BER/CER codecs (X.208)

Platform: any

Classifier: Development Status :: 5 - Production/Stable

Classifier: Environment :: Console

Classifier: Intended Audience :: Developers

Classifier: Intended Audience :: Education

Classifier: Intended Audience :: Information Technology

Classifier: Intended Audience :: System Administrators

Classifier: Intended Audience :: Telecommunications Industry

Classifier: License :: OSI Approved :: BSD License

Classifier: Natural Language :: English

Classifier: Operating System :: OS Independent

Classifier: Programming Language :: Python :: 2

Classifier: Programming Language :: Python :: 2.4

Classifier: Programming Language :: Python :: 2.5

Classifier: Programming Language :: Python :: 2.6

Classifier: Programming Language :: Python :: 2.7

Classifier: Programming Language :: Python :: 3

Classifier: Programming Language :: Python :: 3.2

Classifier: Programming Language :: Python :: 3.3

Classifier: Programming Language :: Python :: 3.4

Classifier: Programming Language :: Python :: 3.5

Classifier: Programming Language :: Python :: 3.6

Classifier: Programming Language :: Python :: 3.7

Classifier: Topic :: Communications

Classifier: Topic :: Software Development :: Libraries :: Python Modules

Found in path(s):

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/PKG-INFO

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1.egg-info/PKG-INFO

No license file was found, but licenses were detected in source scan.

ASN.1 library for Python

========================

.. toctree::
   :maxdepth: 1

Abstract Syntax Notation One (`ASN.1
<http://en.wikipedia.org/wiki/Abstract_Syntax_Notation_1x>`_) is a
technology for exchanging structured data in a universally understood,
hardware agnostic way. Many industrial, security and telephony
applications heavily rely on ASN.1.

The `pyasn1 <https://pypi.org/project/pyasn1/>`_ library implements
ASN.1 support in pure-Python.

What is ASN.1

ASN.1 is a large, arguably over-engineered and extremely old data modelling and
serialisation tool. It is probably among the first serialisation protocols in
the history of computer science and technology.

ASN.1 started its life over 30 years ago as a serialisation mechanism for the first
electronic mail (known as X.400). Later on if was split off the e-mail application
and become a stand-alone tech still being actively supported by its designers
and widely used in industry and technology.

Since then ASN.1 is sort of haunted by its relations with the OSI model -- the
first, unsuccessful, version of the Internet. You can read many interesting
`discussions <https://news.ycombinator.com/item?id=8871453>`_ on that topic.

In the following years, generations of software engineers tackled the serialisation
problem many times. We can see that in Google's `ProtoBuffers <https://developers.google.com/protocol-buffers/>`_
or `FlatBuffers <https://google.github.io/flatbuffers/>`_, for example.
Interestingly, many new takes on binary protocol design do not depart
far from ASN.1 from technical perspective. It's more of a matter of striking
a balance between processing overhead, wire format overhead and human
readability.

Looking at what ASN.1 has to offer, it has three loosely coupled parts:

* Data types: the standard introduces a collection of basic data types
  (integers, bits, strings, arrays and records) that can be used for describing
  arbitrarily complex, nested data structures.

* Serialisation protocols: the above data structures could be converted into a
  series of octets for storage or transmission over the wire as well as
  recovered back into their structured form. The system is fully agnostic

to hardware architectures differences.

* Schema language: ASN.1 data structures could be described in terms
  of a schema language for ASN.1 compiler to turn it into platform-specific
  implementation.

ASN.1 applications

Being an old and generally successful standard, ASN.1 is widely
adopted for many uses. To give you an example, these technologies
use ASN.1 for their data exchange needs:

* Signaling standards for the public switched telephone network (SS7 family)
* Network management standards (SNMP, CMIP)
* Directory standards (X.500 family, LDAP)
* Public Key Infrastructure standards (X.509, etc.)
* PBX control (CSTA)
* IP-based Videoconferencing (H.323 family)
* Biometrics (BIP, CBEFF, ACBio)
* Intelligent transportation (SAE J2735)
* Cellular telephony (GSM, GPRS/EDGE, UMTS, LTE)

ASN.1 gotchas

Apparently, ASN.1 is hard to implement properly. Quality open-source
ASN.1 tools are rare, but ad-hoc implementations are numerous. Judging from the
`statistics <http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=ASN.1>`_ on discovered
security vulnerabilities, many people have implemented ASN.1 parsers
and oftentimes fell victim to its edge cases.

On the bright side, ASN.1 has been around for a long time, it is well understood
and security reviewed.

Documentation

.. toctree::
  :maxdepth: 2

  /pyasn1/contents

Use case

.. toctree::
  :maxdepth: 2

/example-use-case

Download & Install

.. toctree::
  :maxdepth: 2

  /download

Changes

All changes and release history is maintained in changelog. There you
could also download the latest unreleased pyasn1 tarball containing
the latest fixes and improvements.

.. toctree::
  :maxdepth: 1

  /changelog

License

The PyASN1 software is distributed under 2-clause BSD License.

.. toctree::
  :maxdepth: 2

  /license

Getting help

Please, file your `issues <https://github.com/etingof/pyasn1/issues>`_
and `PRs <https://github.com/etingof/pyasn1/pulls>`_ at GitHub.
Alternatively, you could ask for help at
`Stack Overflow <http://stackoverflow.com/questions/tagged/pyasn1>`_
or search
`pyasn1-users <https://lists.sourceforge.net/lists/listinfo/pyasn1-users>`_
mailing list archive.

Books on ASN.1

The pyasn1 implementation is largely based on reading up the following awesome

books:

* `ASN.1 - Communication between heterogeneous systems <http://www.oss.com/asn1/dubuisson.html>`_ by
Olivier Dubuisson
* `ASN.1 Complete <http://www.oss.com/asn1/resources/books-whitepapers-pubs/larmouth-asn1-book.pdf>`_ by
Prof John Larmouth

Here you can get the official standards which is hard to read:

* `ITU standards <http://www.itu.int/ITU-T/studygroups/com17/languages/X.680-X.693-0207w.zip>`_

On the other end of the readability spectrum, here is a quick and sweet write up:

* `A Layman's Guide to a Subset of ASN.1, BER, and DER <ftp://ftp.rsasecurity.com/pub/pkcs/ascii/layman.asc>`_
by Burton S. Kaliski

If you are working with ASN.1, we'd highly recommend reading a proper
book on the subject.

Found in path(s):
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/docs/source/contents.rst
No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/LICENSE.rst
No license file was found, but licenses were detected in source scan.

.. _license:

License
=======

.. include:: ../../LICENSE.rst

Found in path(s):
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/docs/source/license.rst
No license file was found, but licenses were detected in source scan.

ASN.1 library for Python

[![PyPI](https://img.shields.io/pypi/v/pyasn1.svg?maxAge=2592000)](https://pypi.org/project/pyasn1)
[![Python Versions](https://img.shields.io/pypi/pyversions/pyasn1.svg)](https://pypi.org/project/pyasn1/)
[![Build status](https://travis-ci.org/etingof/pyasn1.svg?branch=master)](https://secure.travis-ci.org/etingof/pyasn1)
[![Coverage
Status](https://img.shields.io/codecov/c/github/etingof/pyasn1.svg)](https://codecov.io/github/etingof/pyasn1)
[![GitHub license](https://img.shields.io/badge/license-BSD-
blue.svg)](https://raw.githubusercontent.com/etingof/pyasn1/master/LICENSE.txt)

This is a free and open source implementation of ASN.1 types and codecs
as a Python package. It has been first written to support particular
protocol (SNMP) but then generalized to be suitable for a wide range
of protocols based on
[ASN.1 specification](https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-X.208-198811-W!!PDF-
E&type=items).

Features
----

* Generic implementation of ASN.1 types (X.208)
* Standards compliant BER/CER/DER codecs
* Dumps/loads ASN.1 structures from Python types
* 100% Python, works with Python 2.4 up to Python 3.7
* MT-safe
* Contributed ASN.1 compiler [Asn1ate](https://github.com/kimgr/asn1ate)

Why using pyasn1

ASN.1 solves the data serialisation problem. This solution was
designed long ago by the wise Ancients. Back then, they did not
have the luxury of wasting bits. That is why ASN.1 is designed

to serialise data structures of unbounded complexity into
something compact and efficient when it comes to processing
the data.

That probably explains why many network protocols and file formats
still rely on the 30+ years old technology. Including a number of
high-profile Internet protocols and file formats.

Quite a number of books cover the topic of ASN.1.
[Communication between heterogeneous systems](http://www.oss.com/asn1/dubuisson.html)
by Olivier Dubuisson is one of those high quality books freely
available on the Internet.

The pyasn1 package is designed to help Python programmers tackling
network protocols and file formats at the comfort of their Python
prompt. The tool struggles to capture all aspects of a rather
complicated ASN.1 system and to represent it on the Python terms.

How to use pyasn1

With pyasn1 you can build Python objects from ASN.1 data structures.
For example, the following ASN.1 data structure:

```bash
Record ::= SEQUENCE {
  id      INTEGER,
  room  [0] INTEGER OPTIONAL,
  house [1] INTEGER DEFAULT 0
}
```

Could be expressed in pyasn1 like this:

```python
class Record(Sequence):
  componentType = NamedTypes(
    NamedType('id', Integer()),
    OptionalNamedType(
      'room', Integer().subtype(
        implicitTag=Tag(tagClassContext, tagFormatSimple, 0)
      )
    ),
    DefaultedNamedType(
      'house', Integer(0).subtype(
        implicitTag=Tag(tagClassContext, tagFormatSimple, 1)
      )
    )
```

```
   )
```

It is in the spirit of ASN.1 to take abstract data description
and turn it into a programming language specific form.
Once you have your ASN.1 data structure expressed in Python, you
can use it along the lines of similar Python type (e.g. ASN.1
`SET` is similar to Python `dict`, `SET OF` to `list`):

```python
>>> record = Record()
>>> record['id'] = 123
>>> record['room'] = 321
>>> str(record)
Record:
id=123
room=321
>>>
```

Part of the power of ASN.1 comes from its serialisation features. You
can serialise your data structure and send it over the network.

```python
>>> from pyasn1.codec.der.encoder import encode
>>> substrate = encode(record)
>>> hexdump(substrate)
00000: 30 07 02 01 7B 80 02 01 41
```

Conversely, you can turn serialised ASN.1 content, as received from
network or read from a file, into a Python object which you can
introspect, modify, encode and send back.

```python
>>> from pyasn1.codec.der.decoder import decode
>>> received_record, rest_of_substrate = decode(substrate, asn1Spec=Record())
>>>
>>> for field in received_record:
>>>    print('{} is {}'.format(field, received_record[field]))
id is 123
room is 321
house is 0
>>>
>>> record == received_record
True
>>> received_record.update(room=123)
>>> substrate = encode(received_record)
```

```
>>> hexdump(substrate)
00000: 30 06 02 01 7B 80 01 7B
```

The pyasn1 classes struggle to emulate their Python prototypes (e.g. int,
list, dict etc.). But ASN.1 types exhibit more complicated behaviour.
To make life easier for a Pythonista, they can turn their pyasn1
classes into Python built-ins:

```python
>>> from pyasn1.codec.native.encoder import encode
>>> encode(record)
{'id': 123, 'room': 321, 'house': 0}
```

Or vice-versa -- you can initialize an ASN.1 structure from a tree of
Python objects:

```python
>>> from pyasn1.codec.native.decoder import decode
>>> record = decode({'id': 123, 'room': 321, 'house': 0}, asn1Spec=Record())
>>> str(record)
Record:
id=123
room=321
>>>
```

With ASN.1 design, serialisation codecs are decoupled from data objects,
so you could turn every single ASN.1 object into many different
serialised forms. As of this moment, pyasn1 supports BER, DER, CER and
Python built-ins codecs. The extremely compact PER encoding is expected
to be introduced in the upcoming pyasn1 release.

More information on pyasn1 APIs can be found in the
[documentation](http://snmplabs.com/pyasn1/),
compiled ASN.1 modules for different protocols and file formats
could be found in the pyasn1-modules
[repo](https://github.com/etingof/pyasn1-modules).

How to get pyasn1

The pyasn1 package is distributed under terms and conditions of 2-clause
BSD [license](http://snmplabs.com/pyasn1/license.html). Source code is freely
available as a GitHub [repo](https://github.com/etingof/pyasn1).

You could `pip install pyasn1` or download it from [PyPI](https://pypi.org/project/pyasn1).

If something does not work as expected,
[open an issue](https://github.com/etingof/pyasn1/issues) at GitHub or
post your question [on Stack Overflow](https://stackoverflow.com/questions/ask)
or try browsing pyasn1
[mailing list archives](https://sourceforge.net/p/pyasn1/mailman/pyasn1-users/).

Found in path(s):
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/README.md
No license file was found, but licenses were detected in source scan.

License: http://snmplabs.com/pyasn1/license.html

Found in path(s):
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/__main__.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/ber/decoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/setup.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/cer/test_encoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/univ.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/ber/__main__.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/constraint.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/char.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/cer/__main__.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/debug.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/__main__.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/der/test_decoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/namedtype.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/compat/test_binary.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/der/__main__.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/compat/test_integer.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_tag.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/calling.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_opentype.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_univ.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/native/decoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/cer/test_decoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_char.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/ber/test_encoder.py
* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/native/encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/test_debug.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/error.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/string.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_namedval.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/integer.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_namedtype.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/native/test_encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/tag.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/opentype.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/ber/test_decoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/namedval.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/dateandtime.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/tagmap.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/error.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/cer/encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/ber/encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/compat/__main__.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/native/test_decoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/native/__main__.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/der/encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/octets.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_constraint.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/der/decoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/compat/test_octets.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/base.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/test_useful.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/codec/der/test_encoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/base.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/ber/eoo.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/type/useful.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/tests/type/__main__.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/codec/cer/decoder.py

* /opt/cola/permits/1142175913_1644298289.8/0/pyasn1-0-4-8-1-tar-gz/pyasn1-0.4.8/pyasn1/compat/binary.py

1.33 pyca-cryptography 39.0.2

1.33.1 Available under license :

This software is made available under the terms of *either* of the licenses
found in LICENSE.APACHE or LICENSE.BSD. Contributions to cryptography are made
under the terms of *both* these licenses.

The code used in the OS random engine is derived from CPython, and is licensed under the terms of the PSF License Agreement.
Copyright (c) Individual contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

   1. Redistributions of source code must retain the above copyright notice,
      this list of conditions and the following disclaimer.

   2. Redistributions in binary form must reproduce the above copyright
      notice, this list of conditions and the following disclaimer in the
      documentation and/or other materials provided with the distribution.

   3. Neither the name of PyCA Cryptography nor the names of its contributors
      may be used to endorse or promote products derived from this software
      without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and
   the Individual or Organization ("Licensee") accessing and otherwise using Python
   2.7.12 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
   grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
   analyze, test, perform and/or display publicly, prepare derivative works,
   distribute, and otherwise use Python 2.7.12 alone or in any derivative
   version, provided, however, that PSF's License Agreement and PSF's notice of
   copyright, i.e., "Copyright 2001-2016 Python Software Foundation; All Rights
   Reserved" are retained in Python 2.7.12 alone or in any derivative version
   prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or
   incorporates Python 2.7.12 or any part thereof, and wants to make the
   derivative work available to others as provided herein, then Licensee hereby
   agrees to include in any such work a brief summary of the changes made to Python
   2.7.12.

4. PSF is making Python 2.7.12 available to Licensee on an "AS IS" basis.
 PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF
 EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR
 WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE
 USE OF PYTHON 2.7.12 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 2.7.12
 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF
 MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 2.7.12, OR ANY DERIVATIVE
 THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of
 its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship
 of agency, partnership, or joint venture between PSF and Licensee. This License
 Agreement does not grant permission to use PSF trademarks or trade name in a
 trademark sense to endorse or promote products or services of Licensee, or any
 third party.

8. By copying, installing or otherwise using Python 2.7.12, Licensee agrees
 to be bound by the terms and conditions of this License Agreement.

                          Apache License
                    Version 2.0, January 2004
                   https://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

      "You" (or "Your") shall mean an individual or Legal Entity
      exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.34 pytest-picked 0.4.6

1.34.1 Available under license :

The MIT License (MIT)

Copyright (c) 2018 Ana Paula Gomes

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.35 flake8-annotations-coverage 0.0.6

1.35.1 Available under license :

MIT License

Copyright (c) 2019 BestDoctor

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.36 flake8-annotations-complexity 0.0.7

1.36.1 Available under license :

MIT License

Copyright (c) 2019 BestDoctor

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.37 termcolor-pypi 2.2.0

1.37.1 Available under license :

Copyright (c) 2008-2011 Volvox Development Team

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.38 responses 0.12.1

1.38.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2015 David Cramer

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.39 markdown-it-py 2.2.0

1.39.1 Available under license :

MIT License

Copyright (c) 2020 ExecutableBookProject

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
Copyright (c) 2014 Vitaly Puzrin, Alex Kocharin.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.40 flake8-isort 4.2.0

1.40.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,

under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable
   source code, which must be distributed under the terms of Sections
   1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three
   years, to give any third party, for a charge no more than your
   cost of physically performing source distribution, a complete
   machine-readable copy of the corresponding source code, to be
   distributed under the terms of Sections 1 and 2 above on a medium
   customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer
   to distribute corresponding source code. (This alternative is
   allowed only for noncommercial distribution and only if you
   received the program in object code or executable form with such
   an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

 4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

 5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are

prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

 9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

 10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

                    NO WARRANTY

 11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

 12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

 If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

 To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year> <name of author>

    This program is free software; you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation; either version 2 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License along
    with this program; if not, write to the Free Software Foundation, Inc.,
    51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

 Yoyodyne, Inc., hereby disclaims all copyright interest in the program
 `Gnomovision' (which makes passes at compilers) written by James Hacker.

 <signature of Ty Coon>, 1 April 1989
 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.41 python-dotenv 1.0.0

1.41.1 Available under license :

Copyright (c) 2014, Saurabh Kumar (python-dotenv), 2013, Ted Tieken (django-dotenv-rw), 2013, Jacob Kaplan-Moss (django-dotenv)

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
 this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

- Neither the name of django-dotenv nor the names of its contributors
 may be used to endorse or promote products derived from this software
 without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.42 execnet 1.9.0

1.42.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.43 smmap 5.0.0

1.43.1 Available under license :

Copyright (C) 2010, 2011 Sebastian Thiel and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the async project nor the names of
its contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR

1.44 executing 1.2.0

1.44.1 Available under license :

MIT License

Copyright (c) 2019 Alex Hall

1.45 curlify2 1.0.1

1.45.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: curlify2
Version: 1.0.1
Summary: Library to convert python requests and httpx object to curl command.
Home-page: https://github.com/marcuxyz/curlify2
License: MIT
Keywords: python,curl,requests,curlify,httpx,convert to curl

The library convert python 'requests' and 'httpx' object in curl command. Curlify2 is a enhancement of [curlify]('https://github.com/ofw/curlify').

Installation

To install the library use pip or poetry command, see:

```bash
$ pip install curlify2
```

or poetry:

```bash
$ poetry add curlify2
```

Usage

using **requests** module:

```python
import curlify2
import requests

URL = "https://run.mocky.io/v3/b0f4ffd8-6696-4f90-8bab-4a3bcad9ef3f"

request = requests.get(URL)
curl = curlify2.to_curl(request.request)

print(curl) # curl -X GET -H "User-Agent: python-requests/2.24.0" -H "Accept-Encoding: gzip, deflate" -H
"Accept: */*" -H "Connection: keep-alive" -d 'None' https://run.mocky.io/v3/b0f4ffd8-6696-4f90-8bab-
4a3bcad9ef3f
```

```
```

using **httpx** module:

```python
import curlify2
import httpx

URL = "https://run.mocky.io/v3/b0f4ffd8-6696-4f90-8bab-4a3bcad9ef3f"

request = httpx.get(URL)
curl = curlify2.to_curl(request.request)

print(curl) # curl -X GET -H "User-Agent: python-requests/2.24.0" -H "Accept-Encoding: gzip, deflate" -H
"Accept: */*" -H "Connection: keep-alive" -d 'None' https://run.mocky.io/v3/b0f4ffd8-6696-4f90-8bab-
4a3bcad9ef3f

```

Found in path(s):
* /opt/cola/permits/1343905676_1655169550.7595866/0/curlify2-1-0-1-tar-gz/curlify2-1.0.1/PKG-INFO

1.46 pytest-mock 3.10.0

1.46.1 Available under license :

MIT License

Copyright (c) [2016] [Bruno Oliveira]

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.47 orderedmultidict 1.0.1

1.47.1 Available under license :

Build Amazing Things.

***

Unlicense

This is free and unencumbered software released into the public\
domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute\
this software, either in source code form or as a compiled binary, for any\
purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of\
this software dedicate any and all copyright interest in the software to the\
public domain. We make this dedication for the benefit of the public at\
large and to the detriment of our heirs and successors. We intend this\
dedication to be an overt act of relinquishment in perpetuity of all\
present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF\
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED\
TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A\
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT\
SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR\
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT\
OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION\
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE\
SOFTWARE.

For more information, please refer to <https://unlicense.org/>

1.48 jinja 3.1.2

1.48.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: Jinja2
Version: 3.1.2
Summary: A very fast and expressive template engine.
Home-page: https://palletsprojects.com/p/jinja/

Jinja
=====

Jinja is a fast, expressive, extensible templating engine. Special
placeholders in the template allow writing code similar to Python
syntax. Then the template is passed data to render the final document.

It includes:

-   Template inheritance and inclusion.
-   Define and import macros within templates.
-   HTML templates can use autoescaping to prevent XSS from untrusted
    user input.
-   A sandboxed environment can safely render untrusted templates.
-   AsyncIO support for generating templates and calling async
    functions.
-   I18N support with Babel.
-   Templates are compiled to optimized Python code just-in-time and
    cached, or can be compiled ahead-of-time.
-   Exceptions point to the correct line in templates to make debugging
    easier.
-   Extensible filters, tests, functions, and even syntax.

Jinja's philosophy is that while application logic belongs in Python if possible, it shouldn't make the template designer's job difficult by restricting functionality too much.

Installing

Install and update using `pip`_:

.. code-block:: text

    $ pip install -U Jinja2

.. _pip: https://pip.pypa.io/en/stable/getting-started/

In A Nutshell

.. code-block:: jinja

  {% extends "base.html" %}
  {% block title %}Members{% endblock %}
  {% block content %}

   {% for user in users %}
     {{ user.username }}
   {% endfor %}

  {% endblock %}

Donate

The Pallets organization develops and supports Jinja and other popular packages. In order to grow the community of contributors and users, and allow the maintainers to devote more time to the projects, `please donate today`_.

.. _please donate today: https://palletsprojects.com/donate

Links

- Documentation: https://jinja.palletsprojects.com/
- Changes: https://jinja.palletsprojects.com/changes/
- PyPI Releases: https://pypi.org/project/Jinja2/
- Source Code: https://github.com/pallets/jinja/
- Issue Tracker: https://github.com/pallets/jinja/issues/
- Website: https://palletsprojects.com/p/jinja/
- Twitter: https://twitter.com/PalletsTeam
- Chat: https://discord.gg/pallets

Found in path(s):
* /opt/cola/permits/1357548469_1679936260.2965178/0/jinja2-3-1-2-tar-gz/Jinja2-3.1.2/src/Jinja2.egg-info/PKG-INFO
* /opt/cola/permits/1357548469_1679936260.2965178/0/jinja2-3-1-2-tar-gz/Jinja2-3.1.2/PKG-INFO
No license file was found, but licenses were detected in source scan.

[metadata]
name = Jinja2
version = attr: jinja2.__version__
url = https://palletsprojects.com/p/jinja/
project_urls =
 Donate = https://palletsprojects.com/donate
 Documentation = https://jinja.palletsprojects.com/
 Changes = https://jinja.palletsprojects.com/changes/
 Source Code = https://github.com/pallets/jinja/
 Issue Tracker = https://github.com/pallets/jinja/issues/
 Twitter = https://twitter.com/PalletsTeam
 Chat = https://discord.gg/pallets
license = BSD-3-Clause
license_files = LICENSE.rst
author = Armin Ronacher
author_email = armin.ronacher@active-4.com
maintainer = Pallets
maintainer_email = contact@palletsprojects.com
description = A very fast and expressive template engine.
long_description = file: README.rst
long_description_content_type = text/x-rst
classifiers =
 Development Status :: 5 - Production/Stable
 Environment :: Web Environment
 Intended Audience :: Developers
 License :: OSI Approved :: BSD License
 Operating System :: OS Independent
 Programming Language :: Python
 Topic :: Internet :: WWW/HTTP :: Dynamic Content
 Topic :: Text Processing :: Markup :: HTML

[options]
packages = find:

```
package_dir = = src
include_package_data = True
python_requires = >= 3.7

[options.packages.find]
where = src

[options.entry_points]
babel.extractors =
 jinja2 = jinja2.ext:babel_extract[i18n]

[tool:pytest]
testpaths = tests
filterwarnings =
 error
 ignore:The loop argument:DeprecationWarning:asyncio[.]base_events:542

[coverage:run]
branch = True
source =
 jinja2
 tests

[coverage:paths]
source =
 src
 */site-packages

[flake8]
select = B, E, F, W, B9, ISC
ignore =
 E203
 E501
 E722
 W503
max-line-length = 80
per-file-ignores =
 src/jinja2/__init__.py: F401

[mypy]
files = src/jinja2
python_version = 3.7
show_error_codes = True
disallow_subclassing_any = True
disallow_untyped_calls = True
disallow_untyped_defs = True
disallow_incomplete_defs = True
no_implicit_optional = True
```

```
local_partial_types = True
no_implicit_reexport = True
strict_equality = True
warn_redundant_casts = True
warn_unused_configs = True
warn_unused_ignores = True
warn_return_any = True
warn_unreachable = True

[mypy-jinja2.defaults]
no_implicit_reexport = False

[mypy-markupsafe]
no_implicit_reexport = False

[egg_info]
tag_build =
tag_date = 0
```

Found in path(s):
* /opt/cola/permits/1357548469_1679936260.2965178/0/jinja2-3-1-2-tar-gz/Jinja2-3.1.2/setup.cfg
No license file was found, but licenses were detected in source scan.

Copyright 2007 Pallets

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
    notice, this list of conditions and the following disclaimer in the
    documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF

LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1357548469_1679936260.2965178/0/jinja2-3-1-2-tar-gz/Jinja2-3.1.2/LICENSE.rst

1.49 devtools 0.10.0

1.49.1 Available under license :

The MIT License (MIT)

Copyright (c) 2019 David Zukowski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.50 python-pluggy 1.0.0

1.50.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 holger krekel (rather uses bitbucket/hpk42)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.51 zipp 3.15.0

1.51.1 Available under license :

Copyright Jason R. Coombs

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.52 flake8-annotations 3.0.0

1.52.1 Available under license :

MIT License

Copyright (c) 2019 - Present S. Co1

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.53 python-hyper/rfc3986 1.5.0

1.53.1 Available under license :

Copyright 2014 Ian Cordasco, Rackspace

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.54 pre-commit/pre-commit 3.2.0

1.54.1 Available under license :

Copyright (c) 2012-2013 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2013 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.55 pytest-sugar 0.9.6

1.55.1 Available under license :

Copyright (c) 2013, Teemu <orkkiolento at gmail dot com>
Copyright (c) 2013, Janne Vanhala

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

* The names of the contributors may not be used to endorse or promote products
 derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.56 bandit 1.7.5
1.56.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

1.57 loguru 0.6.0

1.57.1 Available under license :

MIT License

Copyright (c) 2017

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.58 autorepr 0.3.0

1.58.1 Available under license :

tl;dr: all code code is licensed under simplified BSD, unless stated otherwise.

Unless stated otherwise in the source files, all code is copyright 2016 David
Wolever <david@wolever.net>. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

 1. Redistributions of source code must retain the above copyright notice,
 this list of conditions and the following disclaimer.

 2. Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY <COPYRIGHT HOLDER> ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of David Wolever.

1.59 pytest-cov 4.0.0

1.59.1 Available under license :

The MIT License

Copyright (c) 2010 Meme Dough

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.60 timeago 1.0.16

1.60.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016 hustcc

1.61 pytest-xdist 3.2.1

1.61.1 Available under license :

1.62 backoff 2.2.1

1.62.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 litl, LLC.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.63 more-itertools 9.1.0

1.63.1 Available under license :

Copyright (c) 2012 Erik Rose

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.64 starlette 0.26.1

1.64.1 Available under license :

1.65 python-marshmallow 3.19.0

1.65.1 Available under license :

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

marshmallow includes code adapted from Django.

Django License
==============

Copyright (c) Django Software Foundation and individual contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice,
     this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright
     notice, this list of conditions and the following disclaimer in the
     documentation and/or other materials provided with the distribution.

  3. Neither the name of Django nor the names of its contributors may be used
     to endorse or promote products derived from this software without
     specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.66 python-dateutil 2.8.2

1.66.1 Available under license :

Copyright 2017- Paul Ganssle <paul@ganssle.io>
Copyright 2017- dateutil contributors (see AUTHORS file)

  Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The above license applies to all contributions after 2017-12-01, as well as
all contributions that have been re-licensed (see AUTHORS file for the list of
contributors who have re-licensed their code).
----
dateutil - Extensions to the standard Python datetime module.

Copyright (c) 2003-2011 - Gustavo Niemeyer <gustavo@niemeyer.net>
Copyright (c) 2012-2014 - Tomi Pievilinen <tomi.pievilainen@iki.fi>
Copyright (c) 2014-2016 - Yaron de Leeuw <me@jarondl.net>
Copyright (c) 2015-   - Paul Ganssle <paul@ganssle.io>
Copyright (c) 2015-   - dateutil contributors (see AUTHORS file)

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice,
    this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice,
    this list of conditions and the following disclaimer in the documentation
    and/or other materials provided with the distribution.
  * Neither the name of the copyright holder nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The above BSD License Applies to all code, even that also covered by Apache 2.0.

1.67 pytest-timeout 2.1.0

1.67.1 Available under license :

The MIT License

Copyright (C) 2012, 2014 Floris Bruynooghe

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.68 types-pyyaml 6.0.12.8

1.68.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: types-PyYAML
Version: 6.0.12.8
Summary: Typing stubs for PyYAML
Home-page: https://github.com/python/typeshed
License: Apache-2.0 license
Project-URL: GitHub, https://github.com/python/typeshed
Project-URL: Changes, https://github.com/typeshed-
internal/stub_uploader/blob/main/data/changelogs/PyYAML.md
Project-URL: Issue tracker, https://github.com/python/typeshed/issues
Project-URL: Chat, https://gitter.im/python/typing
Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3
Classifier: Typing :: Stubs Only
Description-Content-Type: text/markdown

Typing stubs for PyYAML

This is a PEP 561 type stub package for the `PyYAML` package. It
can be used by type-checking tools like
[mypy](https://github.com/python/mypy/),
[pyright](https://github.com/microsoft/pyright),
[pytype](https://github.com/google/pytype/),
PyCharm, etc. to check code that uses
`PyYAML`. The source for this package can be found at
https://github.com/python/typeshed/tree/main/stubs/PyYAML. All fixes for
types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.
This package was generated from typeshed commit `880c0da4045cd5ff2c29b73429629adf27e49d50`.

Found in path(s):
* /opt/cola/permits/1603644107_1679936724.5851128/0/types-pyyaml-6-0-12-8-tar-gz/types-PyYAML-
6.0.12.8/types_PyYAML.egg-info/PKG-INFO
* /opt/cola/permits/1603644107_1679936724.5851128/0/types-pyyaml-6-0-12-8-tar-gz/types-PyYAML-
6.0.12.8/PKG-INFO

1.69 win32-setctime 1.1.0

1.69.1 Available under license :

MIT License

Copyright (c) 2019 Delgan

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.70 types-python-dateutil 2.8.19.10

1.70.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1

Name: types-python-dateutil

Version: 2.8.19.10

Summary: Typing stubs for python-dateutil

Home-page: https://github.com/python/typeshed

License: Apache-2.0 license

Project-URL: GitHub, https://github.com/python/typeshed

Project-URL: Changes, https://github.com/typeshed-internal/stub_uploader/blob/main/data/changelogs/python-dateutil.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for python-dateutil

This is a PEP 561 type stub package for the `python-dateutil` package. It

can be used by type-checking tools like

[mypy](https://github.com/python/mypy/),

[pyright](https://github.com/microsoft/pyright),

[pytype](https://github.com/google/pytype/),

PyCharm, etc. to check code that uses

`python-dateutil`. The source for this package can be found at

https://github.com/python/typeshed/tree/main/stubs/python-dateutil. All fixes for

types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.

This package was generated from typeshed commit `45f0a5e7e47a23d6be67cfce429c4f2c0de62a29`.

Found in path(s):

* /opt/cola/permits/1603644326_1679525631.6663125/0/types-python-dateutil-2-8-19-10-tar-gz/types-python-dateutil-2.8.19.10/types_python_dateutil.egg-info/PKG-INFO

* /opt/cola/permits/1603644326_1679525631.6663125/0/types-python-dateutil-2-8-19-10-tar-gz/types-python-dateutil-2.8.19.10/PKG-INFO

1.71 pyyaml 6.0

1.71.1 Available under license :

Copyright (c) 2017-2021 Ingy dt Net

Copyright (c) 2006-2016 Kirill Simonov

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.72 entrypoints 0.4

1.72.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Thomas Kluyver and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.73 url-normalize 1.4.3

1.73.1 Available under license :

MIT License

Copyright (c) 2020 Nikolay Panov

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.74 python/importlib_metadata 6.1.0

1.74.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.75 kubernetes 17.17.0

1.75.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2014 The Kubernetes Authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.76 flake8-markdown 0.4.0

1.76.1 Available under license :

MIT License

Copyright (c) 2019 John Franey

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.77 pbr 5.11.1

1.77.1 Available under license :

Copyright (C) 2005 Association of Universities for Research in Astronomy (AURA)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following
   disclaimer in the documentation and/or other materials provided
   with the distribution.

3. The name of AURA and its representatives may not be used to
   endorse or promote products derived from this software without
   specific prior written permission.

THIS SOFTWARE IS PROVIDED BY AURA ``AS IS'' AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL AURA BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

                          Apache License
                     Version 2.0, January 2004
                    http://www.apache.org/licenses/

  TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

  "License" shall mean the terms and conditions for use, reproduction,
  and distribution as defined by Sections 1 through 9 of this document.

  "Licensor" shall mean the copyright owner or entity authorized by
  the copyright owner that is granting the License.

  "Legal Entity" shall mean the union of the acting entity and all
  other entities that control, are controlled by, or are under common
  control with that entity. For the purposes of this definition,
  "control" means (i) the power, direct or indirect, to cause the
  direction or management of such entity, whether by contract or
  otherwise, or (ii) ownership of fifty percent (50%) or more of the
  outstanding shares, or (iii) beneficial ownership of such entity.

  "You" (or "Your") shall mean an individual or Legal Entity
  exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

1.78 pytest-vscodedebug 0.1.0

1.78.1 Available under license :

The MIT License (MIT)

Copyright (c) 2020 Thomas Muguet

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.79 pycparser 2.21

1.79.1 Available under license :

pycparser -- A C parser in Python

Copyright (c) 2008-2020, Eli Bendersky
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,

this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.
* Neither the name of Eli Bendersky nor the names of its contributors may
be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.80 snowball-stemmer 2.2.0

1.80.1 Available under license :

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2004,2005, Richard Boulton
Copyright (c) 2013, Yoshiki Shibukawa
Copyright (c) 2006,2007,2009,2010,2011,2014-2019, Olly Betts
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.
3. Neither the name of the Snowball project nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

1.81 types-setuptools 63.4.1

1.81.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
from abc import abstractmethod
from collections.abc import Iterable, Mapping
from distutils.core import Command as _Command
from typing import Any

from setuptools._deprecation_warning import SetuptoolsDeprecationWarning as SetuptoolsDeprecationWarning
from setuptools.depends import Require as Require
from setuptools.dist import Distribution as Distribution
from setuptools.extension import Extension as Extension

__version__: str

class PackageFinder:
    @classmethod
    def find(cls, where: str = ..., exclude: Iterable[str] = ..., include: Iterable[str] = ...) -> list[str]: ...

class PEP420PackageFinder(PackageFinder): ...

find_packages = PackageFinder.find
find_namespace_packages = PEP420PackageFinder.find

def setup(
    *,
    name: str = ...,
    version: str = ...,
    description: str = ...,
    long_description: str = ...,
    author: str = ...,
    author_email: str = ...,
    maintainer: str = ...,
    maintainer_email: str = ...,
    url: str = ...,
    download_url: str = ...,
    packages: list[str] = ...,
    py_modules: list[str] = ...,
    scripts: list[str] = ...,
    ext_modules: list[Extension] = ...,
    classifiers: list[str] = ...,
    distclass: type[Distribution] = ...,
```

```
      script_name: str = ...,
      script_args: list[str] = ...,
      options: Mapping[str, Any] = ...,
      license: str = ...,
      keywords: list[str] | str = ...,
      platforms: list[str] | str = ...,
      cmdclass: Mapping[str, type[_Command]] = ...,
      data_files: list[tuple[str, list[str]]] = ...,
      package_dir: Mapping[str, str] = ...,
      obsoletes: list[str] = ...,
      provides: list[str] = ...,
      requires: list[str] = ...,
      command_packages: list[str] = ...,
      command_options: Mapping[str, Mapping[str, tuple[Any, Any]]] = ...,
      package_data: Mapping[str, list[str]] = ...,
      include_package_data: bool = ...,
      libraries: list[str] = ...,
      headers: list[str] = ...,
      ext_package: str = ...,
      include_dirs: list[str] = ...,
      password: str = ...,
      fullname: str = ...,
      **attrs: Any,
) -> None: ...


class Command(_Command):
    command_consumes_arguments: bool
    def __init__(self, dist: Distribution, **kw: Any) -> None: ...
    def ensure_string_list(self, option: str | list[str]) -> None: ...
    def reinitialize_command(self, command: _Command | str, reinit_subcommands: int = ..., **kw: Any) ->
_Command: ...
    @abstractmethod
    def initialize_options(self) -> None: ...
    @abstractmethod
    def finalize_options(self) -> None: ...
    @abstractmethod
    def run(self) -> None: ...


class sic(str): ...
```

Found in path(s):
* /opt/cola/permits/1603643796_1679935653.6275518/0/types-setuptools-63-4-1-tar-gz/types-setuptools-
63.4.1/setuptools-stubs/__init__.pyi
No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: types-setuptools
Version: 63.4.1

Summary: Typing stubs for setuptools

Home-page: https://github.com/python/typeshed

License: Apache-2.0 license

Project-URL: GitHub, https://github.com/python/typeshed

Project-URL: Changes, https://github.com/typeshed-
internal/stub_uploader/blob/main/data/changelogs/setuptools.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Platform: UNKNOWN

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for setuptools

This is a PEP 561 type stub package for the `setuptools` package.
It can be used by type-checking tools like mypy, PyCharm, pytype etc. to check code
that uses `setuptools`. The source for this package can be found at
https://github.com/python/typeshed/tree/master/stubs/setuptools. All fixes for
types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/master/README.md for more details.
This package was generated from typeshed commit `5a24bf8e1f1f17561255a290b048e8b50b5282f3`.

Found in path(s):
* /opt/cola/permits/1603643796_1679935653.6275518/0/types-setuptools-63-4-1-tar-gz/types-setuptools-
63.4.1/PKG-INFO
* /opt/cola/permits/1603643796_1679935653.6275518/0/types-setuptools-63-4-1-tar-gz/types-setuptools-
63.4.1/types_setuptools.egg-info/PKG-INFO
No license file was found, but licenses were detected in source scan.

from _typeshed import StrOrBytesPath, SupportsWrite
from collections.abc import Iterable, Mapping
from distutils.cmd import Command
from typing import IO, Any

class DistributionMetadata:
    def __init__(self, path: int | StrOrBytesPath | None = ...) -> None: ...
    name: str | None
    version: str | None
    author: str | None
    author_email: str | None
    maintainer: str | None
    maintainer_email: str | None
    url: str | None
    license: str | None
    description: str | None

```
        long_description: str | None

        keywords: str | list[str] | None

        platforms: str | list[str] | None

        classifiers: str | list[str] | None

        download_url: str | None

        provides: list[str] | None

        requires: list[str] | None

        obsoletes: list[str] | None

        def read_pkg_file(self, file: IO[str]) -> None: ...

        def write_pkg_info(self, base_dir: str) -> None: ...

        def write_pkg_file(self, file: SupportsWrite[str]) -> None: ...

        def get_name(self) -> str: ...

        def get_version(self) -> str: ...

        def get_fullname(self) -> str: ...

        def get_author(self) -> str: ...

        def get_author_email(self) -> str: ...

        def get_maintainer(self) -> str: ...

        def get_maintainer_email(self) -> str: ...

        def get_contact(self) -> str: ...

        def get_contact_email(self) -> str: ...

        def get_url(self) -> str: ...

        def get_license(self) -> str: ...

        def get_licence(self) -> str: ...

        def get_description(self) -> str: ...

        def get_long_description(self) -> str: ...

        def get_keywords(self) -> str | list[str]: ...

        def get_platforms(self) -> str | list[str]: ...

        def get_classifiers(self) -> str | list[str]: ...

        def get_download_url(self) -> str: ...

        def get_requires(self) -> list[str]: ...

        def set_requires(self, value: Iterable[str]) -> None: ...

        def get_provides(self) -> list[str]: ...

        def set_provides(self, value: Iterable[str]) -> None: ...

        def get_obsoletes(self) -> list[str]: ...

        def set_obsoletes(self, value: Iterable[str]) -> None: ...


class Distribution:

        cmdclass: dict[str, type[Command]]

        metadata: DistributionMetadata

        def __init__(self, attrs: Mapping[str, Any] | None = ...) -> None: ...

        def get_option_dict(self, command: str) -> dict[str, tuple[str, str]]: ...

        def parse_config_files(self, filenames: Iterable[str] | None = ...) -> None: ...

        def get_command_obj(self, command: str, create: bool = ...) -> Command | None: ...
```

Found in path(s):
* /opt/cola/permits/1603643796_1679935653.6275518/0/types-setuptools-63-4-1-tar-gz/types-setuptools-
63.4.1/setuptools-stubs/_distutils/dist.pyi

No license file was found, but licenses were detected in source scan.

```python
from collections.abc import Mapping
from distutils.cmd import Command as Command
from distutils.dist import Distribution as Distribution
from distutils.extension import Extension as Extension
from typing import Any

def setup(
    *,
    name: str = ...,
    version: str = ...,
    description: str = ...,
    long_description: str = ...,
    author: str = ...,
    author_email: str = ...,
    maintainer: str = ...,
    maintainer_email: str = ...,
    url: str = ...,
    download_url: str = ...,
    packages: list[str] = ...,
    py_modules: list[str] = ...,
    scripts: list[str] = ...,
    ext_modules: list[Extension] = ...,
    classifiers: list[str] = ...,
    distclass: type[Distribution] = ...,
    script_name: str = ...,
    script_args: list[str] = ...,
    options: Mapping[str, Any] = ...,
    license: str = ...,
    keywords: list[str] | str = ...,
    platforms: list[str] | str = ...,
    cmdclass: Mapping[str, type[Command]] = ...,
    data_files: list[tuple[str, list[str]]] = ...,
    package_dir: Mapping[str, str] = ...,
    obsoletes: list[str] = ...,
    provides: list[str] = ...,
    requires: list[str] = ...,
    command_packages: list[str] = ...,
    command_options: Mapping[str, Mapping[str, tuple[Any, Any]]] = ...,
    package_data: Mapping[str, list[str]] = ...,
    include_package_data: bool = ...,
    libraries: list[str] = ...,
    headers: list[str] = ...,
    ext_package: str = ...,
    include_dirs: list[str] = ...,
    password: str = ...,
    fullname: str = ...,
```

```
    **attrs: Any,
) -> None: ...
def run_setup(script_name: str, script_args: list[str] | None = ..., stop_after: str = ...) -> Distribution: ...
```

Found in path(s):

* /opt/cola/permits/1603643796_1679935653.6275518/0/types-setuptools-63-4-1-tar-gz/types-setuptools-63.4.1/setuptools-stubs/_distutils/core.pyi

1.82 oauthlib/oauthlib 3.2.2

1.82.1 Available under license :

1.83 furl 2.1.3

1.83.1 Available under license :

Build Amazing Things.

***

1.84 markupsafe 2.1.2

1.84.1 Available under license :

No license file was found, but licenses were detected in source scan.

License :: OSI Approved :: BSD License

Found in path(s):
* /opt/cola/permits/1603644060_1679349791.337343/0/sailfishos-mirror-markupsafe-2-1-2-0-gc0254f0-1-tar-gz/sailfishos-mirror-markupsafe-f56c4bc/setup.cfg
No license file was found, but licenses were detected in source scan.

Copyright 2010 Pallets

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
    notice, this list of conditions and the following disclaimer in the
    documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1603644060_1679349791.337343/0/sailfishos-mirror-markupsafe-2-1-2-0-gc0254f0-1-tar-
gz/sailfishos-mirror-markupsafe-f56c4bc/LICENSE.rst

1.85 python-typing-extensions 4.5.0

1.85.1 Available under license :

A. HISTORY OF THE SOFTWARE
==========================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations, which became
Zope Corporation. In 2001, the Python Software Foundation (PSF, see
https://www.python.org/psf/) was formed, a non-profit organization
created specifically to own Python-related Intellectual Property.

Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

Release	Derived from	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes
1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes
1.6	1.5.2	2000	CNRI	no
2.0	1.6	2000	BeOpen.com	no
1.6.1	1.6	2001	CNRI	yes (2)
2.1	2.0+1.6.1	2001	PSF	no
2.0.1	2.0+1.6.1	2001	PSF	yes
2.1.1	2.1+2.0.1	2001	PSF	yes
2.1.2	2.1.1	2002	PSF	yes
2.1.3	2.1.2	2002	PSF	yes
2.2 and above	2.1.1	2001-now	PSF	yes

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under
   the GPL. All Python licenses, unlike the GPL, let you distribute
   a modified version without making your changes open source. The
   GPL-compatible licenses make it possible to combine Python with
   other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible,
   because its license has a choice of law clause. According to
   CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1
   is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's
direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
==

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and

its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022 Python Software Foundation;
All Rights Reserved" are retained in Python alone or in any derivative version
prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National
Research Initiatives, having an office at 1895 Preston White Drive,
Reston, VA 20191 ("CNRI"), and the Individual or Organization
("Licensee") accessing and otherwise using Python 1.6.1 software in
source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a nonexclusive, royalty-free, world-wide
license to reproduce, analyze, test, perform and/or display publicly,
prepare derivative works, distribute, and otherwise use Python 1.6.1
alone or in any derivative version, provided, however, that CNRI's
License Agreement and CNRI's notice of copyright, i.e., "Copyright (c)
1995-2001 Corporation for National Research Initiatives; All Rights
Reserved" are retained in Python 1.6.1 alone or in any derivative
version prepared by Licensee. Alternately, in lieu of CNRI's License
Agreement, Licensee may substitute the following text (omitting the
quotes): "Python 1.6.1 is made available subject to the terms and
conditions in CNRI's License Agreement. This Agreement together with
Python 1.6.1 may be located on the internet using the following
unique, persistent identifier (known as a handle): 1895.22/1013. This
Agreement may also be obtained from a proxy server on the internet
using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python 1.6.1 or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS"
basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. This License Agreement shall be governed by the federal
intellectual property law of the United States, including without
limitation the federal copyright law, and, to the extent such

U.S. federal law does not apply, by the law of the Commonwealth of
Virginia, excluding Virginia's conflict of law provisions.
Notwithstanding the foregoing, with regard to derivative works based
on Python 1.6.1 that incorporate non-separable material that was
previously distributed under the GNU General Public License (GPL), the
law of the Commonwealth of Virginia shall govern this License
Agreement only as to issues arising under or with respect to
Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this
License Agreement shall be deemed to create any relationship of
agency, partnership, or joint venture between CNRI and Licensee. This
License Agreement does not grant permission to use CNRI trademarks or
trade name in a trademark sense to endorse or promote products or
services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying,
installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

    ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam,
The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Stichting Mathematisch
Centrum or CWI not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.86 yarl 1.8.2

1.86.1 Available under license :

Copyright 2016-2021, Andrew Svetlov and aio-libs team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.87 pytest-profiling 1.7.0

1.87.1 Available under license :

MIT License

Copyright (c) 2016 Man AHL

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.88 requests-cache 1.0.0

1.88.1 Available under license :

BSD 2-Clause License

Copyright (c) 2019, Roman Haritonov

Contributors

Thanks to the following individuals for providing pull requests, feedback, and other
contributions that have helped to improve requests-cache:

<!-- ALL-CONTRIBUTORS-LIST:START - Do not remove or modify this section -->
<!-- prettier-ignore-start -->
<!-- markdownlint-disable -->
<table>
 <tbody>
  <tr>
    <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/18150651?v=4?s=100" width="100px;" alt="Aaron ("AJ")
Steers"/>
_{Aaron ("AJ") Steers}
<a href="#ideas-aaronsteers" title="Ideas,
Planning, & Feedback"></td>
    <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/2790092?v=4?s=100" width="100px;" alt="Al Johri"/>
_{Al Johri}
</td>
    <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/27617?v=4?s=100" width="100px;" alt="Alex Morega"/>
_{Alex Morega}
<a href="https://github.com/requests-cache/requests-
cache/commits?author=mgax" title="Documentation"> <a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Amgax" title="Bug reports"></td>
    <td align="center" valign="top" width="14.28%"><img

src="https://avatars.githubusercontent.com/u/287868?v=4?s=100" width="100px;" alt="Alex Sinitsin"/>
_{Alex Sinitsin}
 </td>

   <td align="center" valign="top" width="14.28%">
_{Andrew Kittredge}
</td>

   <td align="center" valign="top" width="14.28%">
_{Andrew Stone}
 </td>

   <td align="center" valign="top" width="14.28%">
_{Antonio Lima}
  </td>

  </tr>
  <tr>
   <td align="center" valign="top" width="14.28%">
_{Boris Danilovich}
 </td>

   <td align="center" valign="top" width="14.28%">
_{Charles}
</td>

   <td align="center" valign="top" width="14.28%">
_{Chris Wesseling}
</td>

   <td align="center" valign="top" width="14.28%">
_{Chris de Graaf}
 </td>

   <td align="center" valign="top" width="14.28%">
_{Christoph Reiter}
 </td>

   <td align="center" valign="top" width="14.28%">
_{Daniel Holth}
</td>

   <td align="center" valign="top" width="14.28%"><img

src="https://avatars.githubusercontent.com/u/128286?v=4?s=100" width="100px;" alt="Daniel Rech"/>
_{Daniel Rech}
 </td>
    </tr>
    <tr>
      <td align="center" valign="top" width="14.28%">
_{David GIS}
</td>
      <td align="center" valign="top" width="14.28%">
_{David Schmidt}
 </td>
      <td align="center" valign="top" width="14.28%">
_{David Stosik}
</td>
      <td align="center" valign="top" width="14.28%">
_{Dragon Dave McKee}
</td>
      <td align="center" valign="top" width="14.28%">
_{Edward Betts}
 </td>
      <td align="center" valign="top" width="14.28%">
_{Elan Ruusame}
   </td>
      <td align="center" valign="top" width="14.28%">
_{Erik Krner}
 </td>
    </tr>
    <tr>
      <td align="center" valign="top" width="14.28%">
_{Evan D}
</td>
      <td align="center" valign="top" width="14.28%">

/>_{FemtoTrader}
   </td>
    <td align="center" valign="top" width="14.28%">
_{Florian Demmer}
 </td>
    <td align="center" valign="top" width="14.28%">
_{GDR!}
</td>
    <td align="center" valign="top" width="14.28%">
_{Garrett Reynolds}
</td>
    <td align="center" valign="top" width="14.28%">
_{Greg Dingle}
</td>
    <td align="center" valign="top" width="14.28%">
_{Guangnan Cheng}
 </td>
  </tr>
  <tr>
    <td align="center" valign="top" width="14.28%">
_{Honza Javorek}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Iftah}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Ilya}
</td>
    <td align="center" valign="top" width="14.28%">
_{J Rob Gant}
 </td>
    <td align="center" valign="top" width="14.28%"><img

src="https://avatars.githubusercontent.com/u/939501?v=4?s=100" width="100px;" alt="Jace Browning"/>
_{Jace Browning}
 </td>

    <td align="center" valign="top" width="14.28%">
_{Jakub Semri}
 </td>

    <td align="center" valign="top" width="14.28%">
_{Jeff O'Neill}
</td>

  </tr>

  <tr>

    <td align="center" valign="top" width="14.28%">
_{Jeremy Douglass}
</td>

    <td align="center" valign="top" width="14.28%">
_{Jessy Williams}
  </td>

    <td align="center" valign="top" width="14.28%">
_{John Vandenberg}
  </td>

    <td align="center" valign="top" width="14.28%">
_{Jonathan Liuti}
</td>

    <td align="center" valign="top" width="14.28%">
_{Jon Janak}
</td>

    <td align="center" valign="top" width="14.28%">
_{Jordan Cook}
      </td>

    <td align="center" valign="top" width="14.28%"><img

src="https://avatars.githubusercontent.com/u/1068245?v=4?s=100" width="100px;" alt="Jrgen Hermann"/>
_{Jrgen Hermann}
 </td>
    </tr>
    <tr>
      <td align="center" valign="top" width="14.28%">
_{Kieran W}
 </td>
      <td align="center" valign="top" width="14.28%">
_{MHellmund}
</td>
      <td align="center" valign="top" width="14.28%">
_{Manuel Eggimann}
 </td>
      <td align="center" valign="top" width="14.28%">
_{Marc Abramowitz}
 </td>
      <td align="center" valign="top" width="14.28%">
_{Marius Gedminas}
 </td>
      <td align="center" valign="top" width="14.28%">
_{Martijn Pieters}
</td>
      <td align="center" valign="top" width="14.28%">
_{Masahiro Wada}
 </td>
    </tr>
    <tr>
      <td align="center" valign="top" width="14.28%">
_{Massimo Santini}
</td>
      <td align="center" valign="top" width="14.28%">
_{Michael Herman}
 <a href="https://github.com/requests-cache/requests-

cache/commits?author=mjhea0" title="Documentation"></td>
    <td align="center" valign="top" width="14.28%">
_{Micha Grny}
</td>
    <td align="center" valign="top" width="14.28%">
_{Micha Nowotka}
</td>
    <td align="center" valign="top" width="14.28%">
_{Mike}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Nate}
</td>
    <td align="center" valign="top" width="14.28%">
_{Nathan Cahill}
</td>
  </tr>
  <tr>
    <td align="center" valign="top" width="14.28%">
_{Nick}
</td>
    <td align="center" valign="top" width="14.28%">
_{Olivier Dalang}
</td>
    <td align="center" valign="top" width="14.28%">
_{Oromion}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Parker Hancock}
      </td>
    <td align="center" valign="top" width="14.28%">
_{Paul Moore}
<a href="#ideas-pfmoore" title="Ideas, Planning, &

Feedback"></td>
    <td align="center" valign="top" width="14.28%">
_{Pawe Krefta}
</td>
    <td align="center" valign="top" width="14.28%">
_{Philipp A.}
</td>
  </tr>
  <tr>
    <td align="center" valign="top" width="14.28%">
_{Rasmus Einarsson}
</td>
    <td align="center" valign="top" width="14.28%">
_{Roderic Day}
</td>
    <td align="center" valign="top" width="14.28%">
_{Roman Haritonov}
      </td>
    <td align="center" valign="top" width="14.28%">
_{Samuel T.}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Sebastian Hffner}
   </td>
    <td align="center" valign="top" width="14.28%">
_{Serhii Chvaliuk}
 </td>
    <td align="center" valign="top" width="14.28%">
_{Simon Biewald}
 </td>

```
    </tr>
    <tr>
    <td align="center" valign="top" width="14.28%"><a href="https://github.com/jseabold"><img
src="https://avatars.githubusercontent.com/u/296164?v=4?s=100" width="100px;" alt="Skipper Seabold"/><br
/><sub><b>Skipper Seabold</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Ajseabold" title="Bug reports"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="http://pathmind.com/"><img
src="https://avatars.githubusercontent.com/u/1197406?v=4?s=100" width="100px;" alt="Slin Lee"/><br
/><sub><b>Slin Lee</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/commits?author=slinlee" title="Documentation"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://www.stavros.io/"><img
src="https://avatars.githubusercontent.com/u/23648?v=4?s=100" width="100px;" alt="Stavros Korokithakis"/><br
/><sub><b>Stavros Korokithakis</b></sub></a><br /><a href="#infra-skorokithakis" title="Infrastructure
(Hosting, Build-Tools, etc)"></a> <a href="#tool-skorokithakis" title="Tools"></a> <a
href="https://github.com/requests-cache/requests-cache/commits?author=skorokithakis"
title="Documentation"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://cheginit.github.io/"><img
src="https://avatars.githubusercontent.com/u/13016644?v=4?s=100" width="100px;" alt="Taher Chegini"/><br
/><sub><b>Taher Chegini</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Acheginit" title="Bug reports"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://github.com/TheTechromancer"><img
src="https://avatars.githubusercontent.com/u/20261699?v=4?s=100" width="100px;" alt="TheTechromancer"/><br
/><sub><b>TheTechromancer</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3ATheTechromancer" title="Bug reports"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://github.com/ValueRaider"><img
src="https://avatars.githubusercontent.com/u/96923577?v=4?s=100" width="100px;" alt="ValueRaider"/><br
/><sub><b>ValueRaider</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/commits?author=ValueRaider" title="Documentation"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://vladimir.panteleev.md/"><img
src="https://avatars.githubusercontent.com/u/160894?v=4?s=100" width="100px;" alt="Vladimir Panteleev"/><br
/><sub><b>Vladimir Panteleev</b></sub></a><br /><a href="#ideas-CyberShadow" title="Ideas, Planning, &
Feedback"></a></td>
    </tr>
    <tr>
    <td align="center" valign="top" width="14.28%"><a href="https://sansec.io/"><img
src="https://avatars.githubusercontent.com/u/1145479?v=4?s=100" width="100px;" alt="Willem de Groot"/><br
/><sub><b>Willem de Groot</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/commits?author=gwillem" title="Code"></a> <a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Agwillem" title="Bug reports"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://github.com/WouterVH"><img
src="https://avatars.githubusercontent.com/u/469509?v=4?s=100" width="100px;" alt="Wouter Vanden
Hove"/><br /><sub><b>Wouter Vanden Hove</b></sub></a><br /><a href="https://github.com/requests-
cache/requests-cache/issues?q=author%3AWouterVH" title="Bug reports"></a></td>
    <td align="center" valign="top" width="14.28%"><a href="https://github.com/YetAnotherNerd"><img
src="https://avatars.githubusercontent.com/u/320738?v=4?s=100" width="100px;" alt="YetAnotherNerd"/><br
/><sub><b>YetAnotherNerd</b></sub></a><br /><a href="https://github.com/requests-cache/requests-
cache/commits?author=YetAnotherNerd" title="Code"></a> <a href="#feature-YetAnotherNerd" title="New
features"></a> <a href="https://github.com/requests-cache/requests-cache/issues?q=author%3AYetAnotherNerd"
```

title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/65560918?v=4?s=100" width="100px;" alt="aaron-mf1"/>
_{aaron-mf1}
<a href="#ideas-aaron-mf1" title="Ideas, Planning, &
Feedback"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/50755629?v=4?s=100" width="100px;" alt="coryairbhb"/>
_{coryairbhb}
<a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Acoryairbhb" title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/972350?v=4?s=100" width="100px;" alt="craig"/>
_{craig}
<a href="https://github.com/requests-cache/requests-
cache/commits?author=craigls" title="Code"> <a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Acraigls" title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/1280390?v=4?s=100" width="100px;" alt="denis-bz"/>
_{denis-bz}
<a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Adenis-bz" title="Bug reports"></td>
     </tr>
     <tr>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/16106844?v=4?s=100" width="100px;" alt="eserdk"/>
_{eserdk}
</td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/11820748?v=4?s=100" width="100px;" alt="girst"/>
_{girst}
<a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Agirst" title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/30716350?v=4?s=100" width="100px;" alt="gorogoroumaru"/>
_{gorogoroumaru}
<a href="https://github.com/requests-cache/requests-
cache/commits?author=gorogoroumaru" title="Code"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/42438361?v=4?s=100" width="100px;" alt="gsalvatella "/>
_{gsalvatella }
<a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Agsalvatella" title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/33844174?v=4?s=100" width="100px;" alt="harvey251"/>
_{harvey251}
<a href="https://github.com/requests-cache/requests-
cache/issues?q=author%3Aharvey251" title="Bug reports"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/6315187?v=4?s=100" width="100px;" alt="masavini"/>
_{masavini}
<a href="https://github.com/requests-cache/requests-
cache/commits?author=masavini" title="Documentation"></td>
       <td align="center" valign="top" width="14.28%"><img
src="https://avatars.githubusercontent.com/u/446561?v=4?s=100" width="100px;" alt="mbarkhau"/>
_{mbarkhau}
<a href="https://github.com/requests-cache/requests-
cache/commits?author=mbarkhau" title="Code"> <a href="https://github.com/requests-cache/requests-
cache/commits?author=mbarkhau" title="Tests"> <a href="#infra-mbarkhau" title="Infrastructure (Hosting,

Build-Tools, etc)"> </td>

  </tr>
  <tr>

    <td align="center" valign="top" width="14.28%">
_{shiftinv}
 </td>

    <td align="center" valign="top" width="14.28%">
_{witionstheme}
</td>

    <td align="center" valign="top" width="14.28%">
_{yassine}
</td>

    <td align="center" valign="top" width="14.28%">
_{}
</td>

  </tr>
 </tbody>
</table>

<!-- markdownlint-restore -->
<!-- prettier-ignore-end -->

<!-- ALL-CONTRIBUTORS-LIST:END -->

This page was made using [all-contributors](https://github.com/all-contributors/all-contributors). See [emoji key](https://allcontributors.org/docs/en/emoji-key) for details on contribution types. If you notice a missing contributor or contribution type, please update it!
{fas}`users` Contributors
```{include} ../../CONTRIBUTORS.md
:start-line: 1
```

1.89 pytest-freezegun 0.4.2

1.89.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017 Tomasz Kontusz

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.90 sniffio 1.3.0

1.90.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
This software is made available under the terms of *either* of the
licenses found in LICENSE.APACHE2 or LICENSE.MIT. Contributions to are
made under the terms of *both* these licenses.

Apache License

<div align="center">
Version 2.0, January 2004

http://www.apache.org/licenses/
</div>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
    distribution, then any Derivative Works that You distribute must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file distributed
    as part of the Derivative Works; within the Source form or
    documentation, if provided along with the Derivative Works; or,
    within a display generated by the Derivative Works, if and
    wherever such third-party notices normally appear. The contents
    of the NOTICE file are for informational purposes only and
    do not modify the License. You may add Your own attribution
    notices within Derivative Works that You distribute, alongside
    or as an addendum to the NOTICE text from the Work, provided
    that such additional attribution notices cannot be construed
    as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.91 pyfiglet 0.8.post1

1.91.1 Available under license :

PyFiglet: An implementation of figlet written in Python

The MIT License (MIT)

Copyright 2007-2018
 Christopher Jones <cjones@insub.org>
 Stefano Rivera <stefano@rivera.za.net>
 Peter Waller <p@pwaller.net>
 And various contributors (see git history).

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the Software), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

1.92 python-slugify 8.0.1

1.92.1 Available under license :

The MIT License

Copyright (c) Val Neekman @ Neekware Inc. http://neekware.com

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.93 pyflakes 2.5.0

1.93.1 Available under license :

Copyright 2005-2011 Divmod, Inc.
Copyright 2013-2014 Florent Xicluna

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.94 toml 0.10.2

1.94.1 Available under license :

The MIT License

Copyright 2013-2019 William Pearson
Copyright 2015-2016 Julien Enselme
Copyright 2016 Google Inc.
Copyright 2017 Samuel Vasko
Copyright 2017 Nate Prewitt
Copyright 2017 Jack Evans
Copyright 2019 Filippo Broggini

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.95 pytest-metadata 2.0.4

1.95.1 Available under license :

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

1.96 cfgv 3.3.1

1.96.1 Available under license :

Copyright (c) 2018 Anthony Sottile

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.97 watchfiles 0.18.1

1.97.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017, 2018, 2019, 2020, 2021, 2022 Samuel Colvin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.98 google-auth 2.16.2

1.98.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
    distribution, then any Derivative Works that You distribute must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file distributed
    as part of the Derivative Works; within the Source form or
    documentation, if provided along with the Derivative Works; or,
    within a display generated by the Derivative Works, if and
    wherever such third-party notices normally appear. The contents
    of the NOTICE file are for informational purposes only and
    do not modify the License. You may add Your own attribution
    notices within Derivative Works that You distribute, alongside
    or as an addendum to the NOTICE text from the Work, provided
    that such additional attribution notices cannot be construed
    as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.99 h11 0.14.0

1.99.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016 Nathaniel J. Smith <njs@pobox.com> and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.100 statesman 1.0.2.post1

1.100.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.101 debugpy 1.6.6

1.101.1 Available under license :

The MIT License (MIT)
Copyright (c) 2016 Red Hat.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

THIRD-PARTY SOFTWARE NOTICES AND INFORMATION
Do Not Translate or Localize

debugpy incorporates third party material from the projects listed below.

1. PyDev.Debugger (https://github.com/fabioz/PyDev.Debugger)
    Includes:File copyright Brainwy Software Ltda.
    Includes:File(s) related to Python, Cpython
    Includes:File authored by Yuli Fitterman
    Includes:File copyright Brainwy software Ltda
    Includes:File with methods from Spyder
    Includes:File(s) related to IPython
    Includes:Files copyright Microsoft Corporation
    Includes:six
    Includes:WinAppDbg
    Includes:XML-RPC client interface for Python

%% PyDev.Debugger NOTICES, INFORMATION, AND LICENSE BEGIN HERE
===
The source code for the PyDev.Debugger files are provided with debugpy, or you may send a check or money order for US $5.00, including the product name (debugpy), the open source component name (PyDev.Debugger) and version number, to: Source Code Compliance Team, Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, USA.

Eclipse Public License, Version 1.0 (EPL-1.0)

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and

unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

===

Includes File copyright Brainwy Software Ltda.

File includes the following notice:

Copyright: Brainwy Software Ltda.

License: EPL.

===

Includes file(s) from Python, Python xreload, Cpython and an ActiveState.com Recipe on "NULL OBJECT

DESIGN PATTERN (PYTHON RECIPE)"

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017 Python Software Foundation; All Rights
Reserved" are retained in Python alone or in any derivative version prepared by
Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee

agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1,

OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. This License Agreement shall be governed by the federal
intellectual property law of the United States, including without
limitation the federal copyright law, and, to the extent such
U.S. federal law does not apply, by the law of the Commonwealth of
Virginia, excluding Virginia's conflict of law provisions.
Notwithstanding the foregoing, with regard to derivative works based
on Python 1.6.1 that incorporate non-separable material that was
previously distributed under the GNU General Public License (GPL), the
law of the Commonwealth of Virginia shall govern this License
Agreement only as to issues arising under or with respect to
Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this
License Agreement shall be deemed to create any relationship of
agency, partnership, or joint venture between CNRI and Licensee. This
License Agreement does not grant permission to use CNRI trademarks or
trade name in a trademark sense to endorse or promote products or
services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying,
installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

    ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (C) 2006-2010 Python Software Foundation

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Stichting Mathematisch
Centrum or CWI not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN

ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

==

Includes File authored by Yuli Fitterman

Copyright (c) Yuli Fitterman

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==

Includes file(s): * Copyright (c) Brainwy software Ltda.
*
* This source code is subject to terms and conditions of the Apache License, Version 2.0. A
* copy of the license can be found in the License.html file at the root of this distribution. If
* you cannot locate the Apache License, Version 2.0, please send an email to
* vspython@microsoft.com. By using this source code in any fashion, you are agreeing to be bound
* by the terms of the Apache License, Version 2.0.
*
* You must not remove this notice, or any other, from this software.

==

Includes file(s): Copyright (c) 2009-2012 Pierre Raybaut

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,

WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
=======================================
Includes file(s) from Ipython

Copyright (c) 2008-2010, IPython Development Team
Copyright (c) 2001-2007, Fernando Perez. <fernando.perez@colorado.edu>
Copyright (c) 2001, Janko Hauser <jhauser@zscout.de>
Copyright (c) 2001, Nathaniel Gray <n8gray@caltech.edu>

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this
list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

Neither the name of the IPython Development Team nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
=======================================
Includes file(s): * Copyright (c) Microsoft Corporation.
*
* This source code is subject to terms and conditions of the Apache License, Version 2.0. A
* copy of the license can be found in the License.html file at the root of this distribution. If
* you cannot locate the Apache License, Version 2.0, please send an email to
* vspython@microsoft.com. By using this source code in any fashion, you are agreeing to be bound
* by the terms of the Apache License, Version 2.0.
*
* You must not remove this notice, or any other, from this software.
=======================================

Includes file(s) from https://pythonhosted.org/six/

Copyright (c) 2010-2018 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
===
Includes WinAppDbg

Copyright (c) 2009-2014, Mario Vilas
All rights reserved.
#
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
#
* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
notice,this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the copyright holder nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.
#
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE

POSSIBILITY OF SUCH DAMAGE.
==
Includes XML-RPC client interface for Python
Copyright (c) 1999-2002 by Secret Labs AB
Copyright (c) 1999-2002 by Fredrik Lundh
#
By obtaining, using, and/or copying this software and/or its
associated documentation, you agree that you have read, understood,
and will comply with the following terms and conditions:
#
Permission to use, copy, modify, and distribute this software and
its associated documentation for any purpose and without fee is
hereby granted, provided that the above copyright notice appears in
all copies, and that both that copyright notice and this permission
notice appear in supporting documentation, and that the name of
Secret Labs AB or the author not be used in advertising or publicity
pertaining to distribution of the software without specific, written
prior permission.
#
SECRET LABS AB AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANT-
ABILITY AND FITNESS. IN NO EVENT SHALL SECRET LABS AB OR THE AUTHOR
BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY
DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE
OF THIS SOFTWARE.
==
Includes https://github.com/vstinner/bytecode e3e77fb690ed05ac171e15694e1c5d0e0dc34e86 - MIT

Copyright (c) 2016 Red Hat.

The MIT License (MIT)
Copyright (c) 2016 Red Hat.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
===
Includes https://github.com/benhoyt/scandir 6ed381881bc2fb9de05804e892eeeeb3601a3af2 - BSD 3-Clause "New"
or "Revised" License

===
END OF PyDev.Debugger NOTICES, INFORMATION, AND LICENSE
/*
*   Apache License
*   Version 2.0, January 2004
*   http://www.apache.org/licenses/
*
* TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
*
* 1. Definitions.
*
*   "License" shall mean the terms and conditions for use, reproduction,

```
 *     and distribution as defined by Sections 1 through 9 of this document.
 *
 *     "Licensor" shall mean the copyright owner or entity authorized by
 *     the copyright owner that is granting the License.
 *
 *     "Legal Entity" shall mean the union of the acting entity and all
 *     other entities that control, are controlled by, or are under common
 *     control with that entity. For the purposes of this definition,
 *     "control" means (i) the power, direct or indirect, to cause the
 *     direction or management of such entity, whether by contract or
 *     otherwise, or (ii) ownership of fifty percent (50%) or more of the
 *     outstanding shares, or (iii) beneficial ownership of such entity.
 *
 *     "You" (or "Your") shall mean an individual or Legal Entity
 *     exercising permissions granted by this License.
 *
 *     "Source" form shall mean the preferred form for making modifications,
 *     including but not limited to software source code, documentation
 *     source, and configuration files.
 *
 *     "Object" form shall mean any form resulting from mechanical
 *     transformation or translation of a Source form, including but
 *     not limited to compiled object code, generated documentation,
 *     and conversions to other media types.
 *
 *     "Work" shall mean the work of authorship, whether in Source or
 *     Object form, made available under the License, as indicated by a
 *     copyright notice that is included in or attached to the work
 *     (an example is provided in the Appendix below).
 *
 *     "Derivative Works" shall mean any work, whether in Source or Object
 *     form, that is based on (or derived from) the Work and for which the
 *     editorial revisions, annotations, elaborations, or other modifications
 *     represent, as a whole, an original work of authorship. For the purposes
 *     of this License, Derivative Works shall not include works that remain
 *     separable from, or merely link (or bind by name) to the interfaces of,
 *     the Work and Derivative Works thereof.
 *
 *     "Contribution" shall mean any work of authorship, including
 *     the original version of the Work and any modifications or additions
 *     to that Work or Derivative Works thereof, that is intentionally
 *     submitted to Licensor for inclusion in the Work by the copyright owner
 *     or by an individual or Legal Entity authorized to submit on behalf of
 *     the copyright owner. For the purposes of this definition, "submitted"
 *     means any form of electronic, verbal, or written communication sent
 *     to the Licensor or its representatives, including but not limited to
 *     communication on electronic mailing lists, source code control systems,
 *     and issue tracking systems that are managed by, or on behalf of, the
```

```
*      Licensor for the purpose of discussing and improving the Work, but
*      excluding communication that is conspicuously marked or otherwise
*      designated in writing by the copyright owner as "Not a Contribution."
*
*      "Contributor" shall mean Licensor and any individual or Legal Entity
*      on behalf of whom a Contribution has been received by Licensor and
*      subsequently incorporated within the Work.
*
*   2. Grant of Copyright License. Subject to the terms and conditions of
*      this License, each Contributor hereby grants to You a perpetual,
*      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*      copyright license to reproduce, prepare Derivative Works of,
*      publicly display, publicly perform, sublicense, and distribute the
*      Work and such Derivative Works in Source or Object form.
*
*   3. Grant of Patent License. Subject to the terms and conditions of
*      this License, each Contributor hereby grants to You a perpetual,
*      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*      (except as stated in this section) patent license to make, have made,
*      use, offer to sell, sell, import, and otherwise transfer the Work,
*      where such license applies only to those patent claims licensable
*      by such Contributor that are necessarily infringed by their
*      Contribution(s) alone or by combination of their Contribution(s)
*      with the Work to which such Contribution(s) was submitted. If You
*      institute patent litigation against any entity (including a
*      cross-claim or counterclaim in a lawsuit) alleging that the Work
*      or a Contribution incorporated within the Work constitutes direct
*      or contributory patent infringement, then any patent licenses
*      granted to You under this License for that Work shall terminate
*      as of the date such litigation is filed.
*
*   4. Redistribution. You may reproduce and distribute copies of the
*      Work or Derivative Works thereof in any medium, with or without
*      modifications, and in Source or Object form, provided that You
*      meet the following conditions:
*
*      (a) You must give any other recipients of the Work or
*          Derivative Works a copy of this License; and
*
*      (b) You must cause any modified files to carry prominent notices
*          stating that You changed the files; and
*
*      (c) You must retain, in the Source form of any Derivative Works
*          that You distribute, all copyright, patent, trademark, and
*          attribution notices from the Source form of the Work,
*          excluding those notices that do not pertain to any part of
*          the Derivative Works; and
*
```

```
*     (d) If the Work includes a "NOTICE" text file as part of its
*         distribution, then any Derivative Works that You distribute must
*         include a readable copy of the attribution notices contained
*         within such NOTICE file, excluding those notices that do not
*         pertain to any part of the Derivative Works, in at least one
*         of the following places: within a NOTICE text file distributed
*         as part of the Derivative Works; within the Source form or
*         documentation, if provided along with the Derivative Works; or,
*         within a display generated by the Derivative Works, if and
*         wherever such third-party notices normally appear. The contents
*         of the NOTICE file are for informational purposes only and
*         do not modify the License. You may add Your own attribution
*         notices within Derivative Works that You distribute, alongside
*         or as an addendum to the NOTICE text from the Work, provided
*         that such additional attribution notices cannot be construed
*         as modifying the License.
*
*     You may add Your own copyright statement to Your modifications and
*     may provide additional or different license terms and conditions
*     for use, reproduction, or distribution of Your modifications, or
*     for any such Derivative Works as a whole, provided Your use,
*     reproduction, and distribution of the Work otherwise complies with
*     the conditions stated in this License.
*
*  5. Submission of Contributions. Unless You explicitly state otherwise,
*     any Contribution intentionally submitted for inclusion in the Work
*     by You to the Licensor shall be under the terms and conditions of
*     this License, without any additional terms or conditions.
*     Notwithstanding the above, nothing herein shall supersede or modify
*     the terms of any separate license agreement you may have executed
*     with Licensor regarding such Contributions.
*
*  6. Trademarks. This License does not grant permission to use the trade
*     names, trademarks, service marks, or product names of the Licensor,
*     except as required for reasonable and customary use in describing the
*     origin of the Work and reproducing the content of the NOTICE file.
*
*  7. Disclaimer of Warranty. Unless required by applicable law or
*     agreed to in writing, Licensor provides the Work (and each
*     Contributor provides its Contributions) on an "AS IS" BASIS,
*     WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
*     implied, including, without limitation, any warranties or conditions
*     of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
*     PARTICULAR PURPOSE. You are solely responsible for determining the
*     appropriateness of using or redistributing the Work and assume any
*     risks associated with Your exercise of permissions under this License.
*
*  8. Limitation of Liability. In no event and under no legal theory,
```

```
 *    whether in tort (including negligence), contract, or otherwise,
 *    unless required by applicable law (such as deliberate and grossly
 *    negligent acts) or agreed to in writing, shall any Contributor be
 *    liable to You for damages, including any direct, indirect, special,
 *    incidental, or consequential damages of any character arising as a
 *    result of this License or out of the use or inability to use the
 *    Work (including but not limited to damages for loss of goodwill,
 *    work stoppage, computer failure or malfunction, or any and all
 *    other commercial damages or losses), even if such Contributor
 *    has been advised of the possibility of such damages.
 *
 *  9. Accepting Warranty or Additional Liability. While redistributing
 *    the Work or Derivative Works thereof, You may choose to offer,
 *    and charge a fee for, acceptance of support, warranty, indemnity,
 *    or other liability obligations and/or rights consistent with this
 *    License. However, in accepting such obligations, You may act only
 *    on Your own behalf and on Your sole responsibility, not on behalf
 *    of any other Contributor, and only if You agree to indemnify,
 *    defend, and hold each Contributor harmless for any liability
 *    incurred by, or claims asserted against, such Contributor by reason
 *    of your accepting any such warranty or additional liability.
 *
 *  END OF TERMS AND CONDITIONS
 *
 *  APPENDIX: How to apply the Apache License to your work.
 *
 *    To apply the Apache License to your work, attach the following
 *    boilerplate notice, with the fields enclosed by brackets "[]"
 *    replaced with your own identifying information. (Don't include
 *    the brackets!)  The text should be enclosed in the appropriate
 *    comment syntax for the file format. We also recommend that a
 *    file or class name and description of purpose be included on the
 *    same "printed page" as the copyright notice for easier
 *    identification within third-party archives.
 *
 *  Copyright [yyyy] [name of copyright owner]
 *
 *  Licensed under the Apache License, Version 2.0 (the "License");
 *  you may not use this file except in compliance with the License.
 *  You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 *  Unless required by applicable law or agreed to in writing, software
 *  distributed under the License is distributed on an "AS IS" BASIS,
 *  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 *  See the License for the specific language governing permissions and
 *  limitations under the License.
```

*/

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other
related items) is being provided by the copyright holders under the following
license. By obtaining, using and/or copying this work, you (the licensee) agree
that you have read, understood, and will comply with the following terms and
conditions.

Permission to copy, modify, and distribute this software and its documentation,
with or without modification, for any purpose and without fee or royalty is
hereby granted, provided that you include the following on ALL copies of the
software and documentation or portions thereof, including modifications:

 1. The full text of this NOTICE in a location viewable to users of the
    redistributed or derivative work.
 2. Any pre-existing intellectual property disclaimers, notices, or terms
    and conditions. If none exist, the W3C Software Short Notice should be
    included (hypertext is preferred, text is permitted) within the body
    of any redistributed or derivative code.
 3. Notice of any changes or modifications to the files, including the date
    changes were made. (We recommend you provide URIs to the location from
    which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE
NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED
TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT
THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY
PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or
publicity pertaining to the software without specific, written prior permission.
Title to copyright in this software and any associated documentation will at
all times remain with copyright holders.

_____

This formulation of W3C's notice and license became active on December 31 2002.
This version removes the copyright ownership notice such that this license can
be used with materials other than those owned by the W3C, reflects that ERCIM
is now a host of the W3C, includes references to this specific dated version of
the license, and removes the ambiguous grant of "use". Otherwise, this version
is the same as the previous version and is written so as to preserve the Free

Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
 However please note future versions of SAX may be covered
 under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
   debugpy

   Copyright (c) Microsoft Corporation
   All rights reserved.

   MIT License

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED *AS IS*, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC

LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
  distributed under this Agreement, and
b) in the case of each subsequent Contributor:
  i) changes to the Program, and
  ii) additions to the Program;

  where such changes and/or additions to the Program originate from and are
  distributed by that particular Contributor. A Contribution 'originates'
  from a Contributor if it was added to the Program by such Contributor
  itself or anyone acting on such Contributor's behalf. Contributions do not
  include additions to the Program which: (i) are separate modules of
  software distributed in conjunction with the Program under their own
  license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
 a) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free copyright license to
   reproduce, prepare derivative works of, publicly display, publicly
   perform, distribute and sublicense the Contribution of such Contributor,
   if any, and such derivative works, in source code and object code form.
 b) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free patent license under
   Licensed Patents to make, use, sell, offer to sell, import and otherwise
   transfer the Contribution of such Contributor, if any, in source code and
   object code form. This patent license shall apply to the combination of
   the Contribution and the Program if, at the time the Contribution is
   added by the Contributor, such addition of the Contribution causes such
   combination to be covered by the Licensed Patents. The patent license
   shall not apply to any other combinations which include the Contribution.

No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
  i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
  ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
  iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
  iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement,

whether expressly, by implication, estoppel or otherwise. All rights in the
Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the
intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year
after the cause of action arose. Each party waives its rights to a jury trial in
any resulting litigation.
Apache Ant
Copyright 1999-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark
Graphics Corp that has been kindly donated to the Apache Software
Foundation.
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH

DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.102 cachetools 5.3.0

1.102.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2022 Thomas Kemmer

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.103 pycodestyle 2.9.1

1.103.1 Available under license :

Copyright 2006-2009 Johann C. Rocholl <johann@rocholl.net>
Copyright 2009-2014 Florent Xicluna <florent.xicluna@gmail.com>
Copyright 2014-2020 Ian Lee <IanLee1521@gmail.com>

Licensed under the terms of the Expat License

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files
(the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of the Software,
and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.104 frozenlist 1.3.3

1.104.1 Available under license :

- Contributors -

Andrew Svetlov

Marcin Konowalczyk

Martijn Pieters

Pau Freixes

Paul Colomiets

Apache License
                        Version 2.0, January 2004
                     http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

     "License" shall mean the terms and conditions for use, reproduction,
     and distribution as defined by Sections 1 through 9 of this document.

     "Licensor" shall mean the copyright owner or entity authorized by
     the copyright owner that is granting the License.

     "Legal Entity" shall mean the union of the acting entity and all
     other entities that control, are controlled by, or are under common
     control with that entity. For the purposes of this definition,
     "control" means (i) the power, direct or indirect, to cause the
     direction or management of such entity, whether by contract or
     otherwise, or (ii) ownership of fifty percent (50%) or more of the
     outstanding shares, or (iii) beneficial ownership of such entity.

     "You" (or "Your") shall mean an individual or Legal Entity
     exercising permissions granted by this License.

     "Source" form shall mean the preferred form for making modifications,
     including but not limited to software source code, documentation
     source, and configuration files.

     "Object" form shall mean any form resulting from mechanical
     transformation or translation of a Source form, including but
     not limited to compiled object code, generated documentation,
     and conversions to other media types.

     "Work" shall mean the work of authorship, whether in Source or
     Object form, made available under the License, as indicated by a
     copyright notice that is included in or attached to the work
     (an example is provided in the Appendix below).

     "Derivative Works" shall mean any work, whether in Source or Object
     form, that is based on (or derived from) the Work and for which the
     editorial revisions, annotations, elaborations, or other modifications
     represent, as a whole, an original work of authorship. For the purposes
     of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
    distribution, then any Derivative Works that You distribute must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file distributed
    as part of the Derivative Works; within the Source form or
    documentation, if provided along with the Derivative Works; or,
    within a display generated by the Derivative Works, if and
    wherever such third-party notices normally appear. The contents
    of the NOTICE file are for informational purposes only and
    do not modify the License. You may add Your own attribution
    notices within Derivative Works that You distribute, alongside
    or as an addendum to the NOTICE text from the Work, provided
    that such additional attribution notices cannot be construed
    as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

1.105 fastapi 0.95.0
1.105.1 Available under license :

1.106 dictdiffer 0.9.0
1.106.1 Available under license :

1.107 flake8-use-fstring 1.4

1.107.1 Available under license :

1.108 marshmallow-polyfield 5.11

1.108.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
  Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
  stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
  that You distribute, all copyright, patent, trademark, and
  attribution notices from the Source form of the Work,
  excluding those notices that do not pertain to any part of
  the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
  distribution, then any Derivative Works that You distribute must
  include a readable copy of the attribution notices contained
  within such NOTICE file, excluding those notices that do not
  pertain to any part of the Derivative Works, in at least one
  of the following places: within a NOTICE text file distributed
  as part of the Derivative Works; within the Source form or
  documentation, if provided along with the Derivative Works; or,
  within a display generated by the Derivative Works, if and
  wherever such third-party notices normally appear. The contents
  of the NOTICE file are for informational purposes only and
  do not modify the License. You may add Your own attribution
  notices within Derivative Works that You distribute, alongside
  or as an addendum to the NOTICE text from the Work, provided
  that such additional attribution notices cannot be construed
  as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
  any Contribution intentionally submitted for inclusion in the Work
  by You to the Licensor shall be under the terms and conditions of
  this License, without any additional terms or conditions.
  Notwithstanding the above, nothing herein shall supersede or modify
  the terms of any separate license agreement you may have executed
  with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
  names, trademarks, service marks, or product names of the Licensor,
  except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

1.109 flake8-plugin-utils 1.3.2

1.109.1 Available under license :

MIT License

Copyright (c) 2019 Afonasev Evgeniy

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.110 flake8-bandit 4.1.1

1.110.1 Available under license :

Copyright (c) 2017 Tyler Wince

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.111 multidict 6.0.4
1.111.1 Available under license :
Copyright 2016-2021 Andrew Svetlov and aio-libs team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.112 pyasn1-modules v0.2.8
1.112.1 Available under license :
Copyright (c) 2005-2019, Ilya Etingof <etingof@gmail.com>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

 * Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

 * Redistributions in binary form must reproduce the above copyright notice,

this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

1.113 pytest-clarity 1.0.1
1.113.1 Available under license :

The MIT License (MIT)

Copyright (c) 2021 Darren Burns

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.114 flake8-builtins 2.1.0

1.114.1 Available under license :

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original

authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

  a) You must cause the modified files to carry prominent notices

stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in
whole or in part contains or is derived from the Program or any
part thereof, to be licensed as a whole at no charge to all third
parties under the terms of this License.

c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide
a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

 3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your

cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

 4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

 5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

 6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to

these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

 7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

 8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

 9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to

address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

  10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

                    NO WARRANTY

  11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

  12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

                    END OF TERMS AND CONDITIONS

            How to Apply These Terms to Your New Programs

  If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year> <name of author>

    This program is free software; you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation; either version 2 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License along
    with this program; if not, write to the Free Software Foundation, Inc.,
    51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

 Yoyodyne, Inc., hereby disclaims all copyright interest in the program
 `Gnomovision' (which makes passes at compilers) written by James Hacker.

 <signature of Ty Coon>, 1 April 1989
 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into

proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.115 python-jsonschema 4.17.3

1.115.1 Available under license :

Copyright (c) 2012 Julian Berman

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2013 Julian Berman

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.116 virtualenv 20.21.0

1.116.1 Available under license :

Copyright (c) 2020-202x The virtualenv developers

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.117 configupdater 3.1.1

1.117.1 Available under license :

ConfigUpdater is licensed under the MIT license; see below for details.

ConfigUpdater includes code derived from the Python standard library, which is
licensed under the Python license, a permissive open source license.
The copyright and license are included at the bottom of this file for compliance
with Python's terms.

The MIT License (MIT)

Copyright (c) 2018 Florian Wilhelm

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2001-present Python Software Foundation; All Rights Reserved

A. HISTORY OF THE SOFTWARE
==========================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations, which became
Zope Corporation. In 2001, the Python Software Foundation (PSF, see
https://www.python.org/psf/) was formed, a non-profit organization
created specifically to own Python-related Intellectual Property.
Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for
the Open Source Definition). Historically, most, but not all, Python
releases have also been GPL-compatible; the table below summarizes
the various releases.

Release	Derived from	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes
1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes

```
1.6        1.5.2    2000    CNRI      no
2.0        1.6      2000    BeOpen.com no
1.6.1      1.6      2001    CNRI      yes (2)
2.1        2.0+1.6.1 2001    PSF       no
2.0.1      2.0+1.6.1 2001    PSF       yes
2.1.1      2.1+2.0.1 2001    PSF       yes
2.1.2      2.1.1    2002    PSF       yes
2.1.3      2.1.2    2002    PSF       yes
2.2 and above  2.1.1    2001-now  PSF       yes
```

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under
    the GPL. All Python licenses, unlike the GPL, let you distribute
    a modified version without making your changes open source. The
    GPL-compatible licenses make it possible to combine Python with
    other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible,
    because its license has a choice of law clause. According to
    CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1
    is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's
direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
===

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020 Python Software Foundation;
All Rights Reserved" are retained in Python alone or in any derivative version
prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,

provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National
Research Initiatives, having an office at 1895 Preston White Drive,
Reston, VA 20191 ("CNRI"), and the Individual or Organization
("Licensee") accessing and otherwise using Python 1.6.1 software in
source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a nonexclusive, royalty-free, world-wide
license to reproduce, analyze, test, perform and/or display publicly,
prepare derivative works, distribute, and otherwise use Python 1.6.1
alone or in any derivative version, provided, however, that CNRI's

License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or

services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying,
installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam,
The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Stichting Mathematisch
Centrum or CWI not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.118 asttokens 2.2.1
1.118.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.119 pkgutil_resolve_name 1.3.10

1.119.1 Available under license :

The MIT License (MIT)

Copyright (c) 2020 Thomas Grainger.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Portions of this pkgutil-resolve-name Software may utilize the following copyrighted material, the use of which is
hereby acknowledged.

cpython: https://github.com/python/cpython/tree/1ed61617a4a6632905ad6a0b440cd2cafb8b6414

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020 Python Software Foundation;
All Rights Reserved" are retained in Python alone or in any derivative version

prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

1.120 uvloop 0.17.0

1.120.1 Available under license :

The MIT License

Copyright (C) 2016-present the uvloop authors and contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (C) 2016-present the uvloop authors and contributors.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

1.121 pytest-randomly 3.12.0

1.121.1 Available under license :

1.122 gitdb 4.0.10

1.122.1 Available under license :

Additional Licenses

The files at
gitdb/test/fixtures/packs/pack-11fdfa9e156ab73caae3b6da867192221f2089c2.idx
and
gitdb/test/fixtures/packs/pack-11fdfa9e156ab73caae3b6da867192221f2089c2.pack
are licensed under GNU GPL as part of the git source repository,
see http://en.wikipedia.org/wiki/Git_%28software%29 for more information.

They are not required for the actual operation, which is why they are not found
in the distribution package.

1.123 flatten-dict 0.4.2

1.123.1 Available under license :

MIT License

Copyright (c) 2019 Ian Lin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.124 identify 2.5.21

1.124.1 Available under license :

Copyright (c) 2017 Chris Kuehl, Anthony Sottile

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.125 authlib 1.2.0

1.125.1 Available under license :

```
============================
```
Authlib - Terms and conditions
```
============================
```

1. **Preamble**:

This Agreement governs the relationship between "YOU" (hereinafter: Licensee) and Hsiaoming Yang (hereinafter: Licensor). This Agreement sets the terms, rights, restrictions and obligations on using Authlib (hereinafter: The Software) created and owned by Licensor, as detailed herein

2. **License Grant**:

Licensor hereby grants Licensee a Personal, Non-assignable & non-transferable, Commercial, Royalty free, Including the rights to create but not distribute derivative works, Non-exclusive license, all with accordance with the terms set forth and other legal restrictions set forth in 3rd party software used while running Software.

2.1. **Limited**: Licensee may use Software for the purpose of:

2.1.1. Running Software on Licensees Website[s] and Server[s];
2.1.2. Modify Software to suit Licensees needs and specifications.

2.2. **Non Assignable & Non-Transferable**: Licensee may not assign or transfer his rights and duties under this license.

2.3. **Commercial, Royalty Free**: Licensee may use Software for any purpose, including paid-services, without any royalties

2.4. **Including the Right to Create Derivative Works**: Licensee may create derivative works based on Software, including amending Softwares source code, modifying it, integrating it into a larger work or removing portions of Software, as long as no distribution of the derivative works is made

3. **Term & Termination**:

The Term of this license shall be until terminated. Licensor may terminate this Agreement, including Licensees license in the case where Licensee:

3.1. became insolvent or otherwise entered into any liquidation process;

3.2. exported The Software to any jurisdiction where licensor may not enforce his rights under this agreements in;

3.3. Licensee was in breach of any of this license's terms and conditions and such breach was not cured, immediately upon notification;

3.4. Licensee in breach of any of the terms of clause 2 to this license;

3.5. Licensee otherwise entered into any arrangement which caused Licensor to be unable to enforce his rights under this License.

4. **Payment**:

In consideration of the License granted under clause 2, Licensee shall pay Licensor a fee, via Credit-Card, PayPal or any other mean which Licensor may deem adequate. Failure to perform payment shall construe as material breach of this Agreement.

5. **Upgrades, Updates and Fixes**:

Licensor may provide Licensee, from time to time, with Upgrades, Updates or Fixes, as detailed herein and according to his sole discretion. Licensee hereby warrants to keep The Software up-to-date and install all relevant updates and fixes. Licensor shall provide any update or Fix free of charge; however, nothing in this Agreement shall require Licensor to provide Updates or Fixes.

5.1. **Upgrades**: for the purpose of this license, an Upgrade shall be a material amendment in The Software, which contains new features and or major performance improvements and shall be marked as a new version number. For example, The Software under version 1.X.X, an upgrade shall commence under number 2.0.0.

5.2. **Updates**: for the purpose of this license, an update shall be a minor amendment in The Software, which may contain new features or minor improvements and shall be marked as a new sub-version number. For example, The Software under version 1.1.X, an upgrade shall commence under number 1.2.0.

5.3. **Fix**: for the purpose of this license, a fix shall be a minor amendment in The Software, intended to remove bugs or alter minor features which impair the The Software's functionality. A fix shall be marked as a new sub-sub-version number. For example, Software under version 1.1.1, an upgrade shall commence under number 1.1.2.

6. **Support**:

Software is provided under an AS-IS basis and without any support, updates or maintenance. Nothing in this Agreement shall require Licensor to provide Licensee with support or fixes to any bug, failure, mis-performance or other defect in The Software.

6.1. **Bug Notification**: Licensee may provide Licensor of details regarding

any bug, defect or failure in The Software promptly and with no delay from such event; Licensee shall comply with Licensor's request for information regarding bugs, defects or failures and furnish him with information, screenshots and try to reproduce such bugs, defects or failures.

6.2. **Feature Request**: Licensee may request additional features in Software, provided, however, that

(i) Licensee shall waive any claim or right in such feature should feature be developed by Licensor;

(ii) Licensee shall be prohibited from developing the feature, or disclose such feature request, or feature, to any 3rd party directly competing with Licensor or any 3rd party which may be, following the development of such feature, in direct competition with Licensor;

(iii) Licensee warrants that feature does not infringe any 3rd party patent, trademark, trade-secret or any other intellectual property right; and

(iv) Licensee developed, envisioned or created the feature solely by himself.

7. **Liability**:

To the extent permitted under Law, The Software is provided under an AS-IS basis. Licensor shall never, and without any limit, be liable for any damage, cost, expense or any other payment incurred by Licensee as a result of Softwares actions, failure, bugs and/or any other interaction between The Software and Licensees end-equipment, computers, other software or any 3rd party, end-equipment, computer or services. Moreover, Licensor shall never be liable for any defect in source code written by Licensee when relying on The Software or using The Softwares source code.

8. **Warranty**:

8.1. **Intellectual Property**: Licensor hereby warrants that The Software does not violate or infringe any 3rd party claims in regards to intellectual property, patents and/or trademarks and that to the best of its knowledge no legal action has been taken against it for any infringement or violation of any 3rd party intellectual property rights.

8.2. **No-Warranty**: The Software is provided without any warranty; Licensor hereby disclaims any warranty that The Software shall be error free, without defects or code which may cause damage to Licensees computers or to Licensee, and that Software shall be functional. Licensee shall be solely liable to any damage, defect or loss incurred as a result of operating software and undertake the risks contained in running The Software on Licenses Server[s] and Website[s].

8.3. **Prior Inspection**: Licensee hereby states that he inspected The Software thoroughly and found it satisfactory and adequate to his needs, that it does not interfere with his regular operation and that it does meet the standards and scope of his computer systems and architecture. Licensee found that The Software interacts with his development, website and server environment and that it does not infringe any of End User License Agreement of any software Licensee may use in performing his

services. Licensee hereby waives any claims regarding The Software's incompatibility, performance, results and features, and warrants that he inspected the The Software.

9. **No Refunds**:

  Licensee warrants that he inspected The Software according to above clauses and that it is adequate to his needs. Accordingly, as The Software is intangible goods, Licensee shall not be, ever, entitled to any refund, rebate, compensation or restitution for any reason whatsoever, even if The Software contains material flaws.

10. **Indemnification**:

  Licensee hereby warrants to hold Licensor harmless and indemnify Licensor for any lawsuit brought against it in regards to Licensees use of The Software in means that violate, breach or otherwise circumvent this license, Licensor's intellectual property rights or Licensor's title in The Software. Licensor shall promptly notify Licensee in case of such legal action and request Licensees consent prior to any settlement in relation to such lawsuit or claim.

11. **Governing Law, Jurisdiction**:

  Licensee hereby agrees not to initiate class-action lawsuits against Licensor in relation to this license and to compensate Licensor for any legal fees, cost or attorney fees should any claim brought by Licensee against Licensor be denied, in part or in full.

BSD 3-Clause License

Copyright (c) 2017, Hsiaoming Yang
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.126 freezegun 1.2.2

1.126.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2012 Steve Pulec

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.127 pydocstyle 6.3.0
1.127.1 Available under license :
Copyright (c) 2012 GreenSteam, <http://greensteam.dk/>

Copyright (c) 2014-2020 Amir Rachum, <http://amir.rachum.com/>

Copyright (c) 2020 Sambhav Kothari, <https://github.com/samj1912>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.128 nodeenv 1.7.0

1.128.1 Available under license :

Copyright (c) 2011, Eugene Kalinin.

Some rights reserved.

Redistribution and use in source and binary forms of the software as well as documentation, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
 notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
 copyright notice, this list of conditions and the following
 disclaimer in the documentation and/or other materials provided
 with the distribution.

* The names of the contributors may not be used to endorse or
 promote products derived from this software without specific
 prior written permission.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE AND DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.129 packaging 23.0

1.129.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

   END OF TERMS AND CONDITIONS

documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.130 yamlpath 3.7.0

1.130.1 Available under license :

ISC License

Copyright (c) 2001-2020, William W. Kimball Jr. MBA MSIS

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.131 ronnypfannschmidt/iniconfig 2.0.0

1.131.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

1.132 coverage 7.2.2

1.132.1 Available under license :

Copyright 2001 Gareth Rees. All rights reserved.
Copyright 2004-2023 Ned Batchelder. All rights reserved.

Except where noted otherwise, this software is licensed under the Apache
License, Version 2.0 (the "License"); you may not use this work except in
compliance with the License. You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

                    Apache License
                Version 2.0, January 2004
              http://www.apache.org/licenses/

    TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

    1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Coverage.py was originally written by Gareth Rees, and since 2004 has been
extended and maintained by Ned Batchelder.

Other contributions, including writing code, updating docs, and submitting
useful bug reports, have been made by:

Abdeali Kothari
Adi Roiban
Agbonze O. Jeremiah
Albertas Agejevas
Aleksi Torhamo
Alex Gaynor
Alex Groce
Alex Sandro
Alexander Todorov
Alexander Walters
Alpha Chen
Ammar Askar
Andrew Hoos
Anthony Sottile
Arcadiy Ivanov
Aron Griffis
Artem Dayneko
Arthur Deygin
Arthur Rio
Ben Carlsson
Ben Finney
Benjamin Parzella
Benjamin Schubert
Bernt Gbor
Bill Hart
Bradley Burns
Brandon Rhodes
Brett Cannon
Bruno P. Kinoshita
Buck Evan
Calen Pennington

Carl Friedrich Bolz-Tereick

Carl Gieringer

Catherine Proulx

Chris Adams

Chris Jerdonek

Chris Rose

Chris Warrick

Christian Heimes

Christine Lytwynec

Christoph Blessing

Christoph Zwerschke

Clment Pit-Claudel

Conrad Ho

Cosimo Lupo

Dan Hemberger

Dan Riti

Dan Wandschneider

Danek Duvall

Daniel Hahler

Danny Allen

David Christian

David MacIver

David Stanek

David Szotten

Detlev Offenbach

Devin Jeanpierre

Dirk Thomas

Dmitry Shishov

Dmitry Trofimov

Eduardo Schettino

Edward Loper

Eli Skeggs

Emil Madsen

ric Larivire

Federico Bond

Felix Horvat

Frazer McLean

Geoff Bache

George Paci

George Song

George-Cristian Brzan

Greg Rogers

Guido van Rossum

Guillaume Chazarain

Hugo van Kemenade

Ian Moore

Ilia Meerovich

Imri Goldberg

Ionel Cristian Mrie

Ivan Ciuvalschii

J. M. F. Tsang

JT Olds

Jerin Peter George

Jessamyn Smith

Joe Doherty

Joe Jevnik

Jon Chappell

Jon Dufresne

Joseph Tate

Josh Williams

Judson Neer

Julian Berman

Julien Voisin

Justas Sadzeviius

Kjell Braden

Krystian Kichewko

Kyle Altendorf

Lars Hupfeldt Nielsen

Leonardo Pistone

Lex Berezhny

Loc Dachary

Lorenzo Mic

Manuel Jacob

Marc Abramowitz

Marc Legendre

Marcelo Trylesinski

Marcus Cobden

Marius Gedminas

Mark van der Wal

Martin Fuzzey

Matt Bachmann

Matthew Boehm

Matthew Desmarais

Matus Valo

Max Linke

Michael Krebs

Micha Bultrowicz

Micha Grny

Mickie Betz

Mike Fiedler

Naveen Yadav

Nathan Land

Nikita Bloshchanevich

Nils Kattenbeck

Noel O'Boyle

Olivier Grisel

Ori Avtalion

Pankaj Pandey

Pablo Carballo

Patrick Mezard

Peter Baughman

Peter Ebden

Peter Portante

Reya B

Rodrigue Cloutier

Roger Hu

Ross Lawley

Roy Williams

Russell Keith-Magee

Salvatore Zagaria

Sandra Martocchia

Scott Belden

Sebastin Ramrez

Sergey B Kirpichev

Sigve Tjora

Simon Willison

Stan Hu

Stefan Behnel

Stephan Deibel

Stephan Richter

Stephen Finucane

Steve Dower

Steve Leonard

Steve Oswald

Steve Peak

Sviatoslav Sydorenko

S. Y. Lee

Teake Nutma

Ted Wexler

Thijs Triemstra

Thomas Grainger

Titus Brown

Valentin Lab

Vince Salvino

Ville Skytt

Xie Yanbo

Yilei "Dolee" Yang

Yury Selivanov

Zac Hatfield-Dodds

Zooko Wilcox-O'Hearn

1.133 text-unidecode 1.3

1.133.1 Available under license :

text-unidecode is a free software; you can redistribute
it and/or modify it under the terms of either:

* GPL or GPLv2+ (see https://www.gnu.org/licenses/license-list.html#GNUGPL), or
* Artistic License - see below:

                    The "Artistic License"

                         Preamble

The intent of this document is to state the conditions under which a
Package may be copied, such that the Copyright Holder maintains some
semblance of artistic control over the development of the package,
while giving the users of the package the right to use and distribute
the Package in a more-or-less customary fashion, plus the right to make
reasonable modifications.

Definitions:

    "Package" refers to the collection of files distributed by the
    Copyright Holder, and derivatives of that collection of files
    created through textual modification.

    "Standard Version" refers to such a Package if it has not been
    modified, or has been modified in accordance with the wishes
    of the Copyright Holder as specified below.

    "Copyright Holder" is whoever is named in the copyright or
    copyrights for the package.

    "You" is you, if you're thinking about copying or distributing
    this Package.

    "Reasonable copying fee" is whatever you can justify on the
    basis of media cost, duplication charges, time of people involved,
    and so on. (You will not be required to justify it to the
    Copyright Holder, but only to the computing community at large
    as a market that must bear the fee.)

    "Freely Available" means that no fee is charged for the item
    itself, though there may be fees involved in handling the item.
    It also means that recipients of the item may redistribute it
    under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

   b) use the modified Package only within your corporation or organization.

   c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

   a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

   b) accompany the distribution with the machine-readable source of the Package with your modifications.

   c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However,

you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End

1.134 dpath-python 2.1.4

1.134.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 Andrew Kesterson <andrew@aklabs.net>, Caleb Case <caleb.case@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.135 async-timeout 4.0.2

1.135.1 Available under license :

Copyright 2016-2020 aio-libs collaboration.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.136 eradicate 2.2.0

1.136.1 Available under license :

No license file was found, but licenses were detected in source scan.

Permission is hereby granted, free of charge, to any person obtaining

a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

Found in path(s):
* /opt/cola/permits/1603644454_1679349763.0190315/0/wemake-services-eradicate-2-2-0-0-g3cdf236-tar-gz/wemake-services-eradicate-29b6f15/eradicate.py
* /opt/cola/permits/1603644454_1679349763.0190315/0/wemake-services-eradicate-2-2-0-0-g3cdf236-tar-gz/wemake-services-eradicate-29b6f15/eradicate
No license file was found, but licenses were detected in source scan.

'License :: OSI Approved :: MIT License',

Found in path(s):
* /opt/cola/permits/1603644454_1679349763.0190315/0/wemake-services-eradicate-2-2-0-0-g3cdf236-tar-gz/wemake-services-eradicate-29b6f15/setup.py

1.137 flake 5.0.4

1.137.1 Available under license :

Project created by Tarek Ziad.

Contributors (by order of appearance) :

- Tams Gulcsi
- Nicolas Dumazet
- Stefan Scherfke
- Chris Adams
- Ben Bass
- Ask Solem
- Steven Kryskalla
- Gustavo Picon
- Jannis Leidel
- Miki Tebeka
- David Cramer
- Peter Teichman
- Ian Cordasco
- Oleg Broytman
- Marc Labb
- Bruno Miguel Custdio
- Florent Xicluna
- Austin Morton
- Michael McNeil Forbes
- Christian Long

- Tyrel Souza

- Corey Farwell

- Michael Penkov

- Anthony Sottile

== Flake8 License (MIT) ==

Copyright (C) 2011-2013 Tarek Ziade <tarek@ziade.org>

Copyright (C) 2012-2016 Ian Cordasco <graffatcolmingov@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.138 nitpick 0.32.0

1.138.1 Available under license :

plugins:
 git-legal:
   enabled: true
[nitpick.meta]
name = "Git.legal - CodeClimate Community Edition"
url = "https://github.com/kmewhort/git.legal-codeclimate"

[".codeclimate.yml".plugins.git-legal] # https://docs.codeclimate.com/docs/git-legal
enabled = true
MIT License

Copyright (c) 2018 Wagner Augusto Andreoli

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.139 mypy 1.1.1

1.139.1 Available under license :

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The "typeshed" project is licensed under the terms of the Apache license, as
reproduced below.

= = = = =

Apache License
                Version 2.0, January 2004
                http://www.apache.org/licenses/

 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

 1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
   distribution, then any Derivative Works that You distribute must
   include a readable copy of the attribution notices contained
   within such NOTICE file, excluding those notices that do not
   pertain to any part of the Derivative Works, in at least one
   of the following places: within a NOTICE text file distributed
   as part of the Derivative Works; within the Source form or
   documentation, if provided along with the Derivative Works; or,
   within a display generated by the Derivative Works, if and
   wherever such third-party notices normally appear. The contents
   of the NOTICE file are for informational purposes only and
   do not modify the License. You may add Your own attribution
   notices within Derivative Works that You distribute, alongside
   or as an addendum to the NOTICE text from the Work, provided
   that such additional attribution notices cannot be construed
   as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

= = = = =

Parts of typeshed are licensed under different licenses (like the MIT
license), reproduced below.

= = = = =

The MIT License

Copyright (c) 2015 Jukka Lehtosalo and contributors

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

= = = = =
Mypy (and mypyc) are licensed under the terms of the MIT license, reproduced below.

= = = = =

The MIT License

Copyright (c) 2012-2022 Jukka Lehtosalo and contributors
Copyright (c) 2015-2022 Dropbox, Inc.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

= = = = =

Portions of mypy and mypyc are licensed under different licenses.
The files
mypyc/lib-rt/pythonsupport.h, mypyc/lib-rt/getargs.c and
mypyc/lib-rt/getargsfast.c are licensed under the PSF 2 License, reproduced
below.

= = = = =

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012 Python Software Foundation; All Rights Reserved" are retained in Python
alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and

otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National
Research Initiatives, having an office at 1895 Preston White Drive,
Reston, VA 20191 ("CNRI"), and the Individual or Organization
("Licensee") accessing and otherwise using Python 1.6.1 software in
source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a nonexclusive, royalty-free, world-wide
license to reproduce, analyze, test, perform and/or display publicly,
prepare derivative works, distribute, and otherwise use Python 1.6.1

alone or in any derivative version, provided, however, that CNRI's
License Agreement and CNRI's notice of copyright, i.e., "Copyright (c)
1995-2001 Corporation for National Research Initiatives; All Rights
Reserved" are retained in Python 1.6.1 alone or in any derivative
version prepared by Licensee. Alternately, in lieu of CNRI's License
Agreement, Licensee may substitute the following text (omitting the
quotes): "Python 1.6.1 is made available subject to the terms and
conditions in CNRI's License Agreement. This Agreement together with
Python 1.6.1 may be located on the Internet using the following
unique, persistent identifier (known as a handle): 1895.22/1013. This
Agreement may also be obtained from a proxy server on the Internet
using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python 1.6.1 or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS"
basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. This License Agreement shall be governed by the federal
intellectual property law of the United States, including without
limitation the federal copyright law, and, to the extent such
U.S. federal law does not apply, by the law of the Commonwealth of
Virginia, excluding Virginia's conflict of law provisions.
Notwithstanding the foregoing, with regard to derivative works based
on Python 1.6.1 that incorporate non-separable material that was
previously distributed under the GNU General Public License (GPL), the
law of the Commonwealth of Virginia shall govern this License
Agreement only as to issues arising under or with respect to
Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this
License Agreement shall be deemed to create any relationship of
agency, partnership, or joint venture between CNRI and Licensee. This
License Agreement does not grant permission to use CNRI trademarks or

trade name in a trademark sense to endorse or promote products or
services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying,
installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

  ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam,
The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Stichting Mathematisch
Centrum or CWI not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.140 types-freezegun 1.1.10

1.140.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: types-freezegun
Version: 1.1.10
Summary: Typing stubs for freezegun
Home-page: https://github.com/python/typeshed
License: Apache-2.0 license
Project-URL: GitHub, https://github.com/python/typeshed
Project-URL: Changes, https://github.com/typeshed-
internal/stub_uploader/blob/main/data/changelogs/freezegun.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Platform: UNKNOWN

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for freezegun

This is a PEP 561 type stub package for the `freezegun` package.
It can be used by type-checking tools like mypy, PyCharm, pytype etc. to check code
that uses `freezegun`. The source for this package can be found at
https://github.com/python/typeshed/tree/master/stubs/freezegun. All fixes for
types and metadata should be contributed there.

Note: The `freezegun` package includes type annotations or type stubs
since version 1.2.1. Please uninstall the `types-freezegun`
package if you use this or a newer version.

See https://github.com/python/typeshed/blob/master/README.md for more details.
This package was generated from typeshed commit `81fd55a885e678559471710f249c7e27d02ec212`.

Found in path(s):
* /opt/cola/permits/1603644551_1679936260.3702052/0/types-freezegun-1-1-10-tar-gz/types-freezegun-1.1.10/types_freezegun.egg-info/PKG-INFO
* /opt/cola/permits/1603644551_1679936260.3702052/0/types-freezegun-1-1-10-tar-gz/types-freezegun-1.1.10/PKG-INFO

1.141 pytest-asyncio 0.20.3
1.141.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.142 click-python-command-line-utility 8.1.3

1.142.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1

Name: click

Version: 8.1.3

Summary: Composable command line interface toolkit

Home-page: https://palletsprojects.com/p/click/

Author: Armin Ronacher

Author-email: armin.ronacher@active-4.com

Maintainer: Pallets

Maintainer-email: contact@palletsprojects.com

License: BSD-3-Clause

Project-URL: Donate, https://palletsprojects.com/donate

Project-URL: Documentation, https://click.palletsprojects.com/

Project-URL: Changes, https://click.palletsprojects.com/changes/

Project-URL: Source Code, https://github.com/pallets/click/

Project-URL: Issue Tracker, https://github.com/pallets/click/issues/

Project-URL: Twitter, https://twitter.com/PalletsTeam

Project-URL: Chat, https://discord.gg/pallets

Platform: UNKNOWN

Classifier: Development Status :: 5 - Production/Stable

Classifier: Intended Audience :: Developers

Classifier: License :: OSI Approved :: BSD License

Classifier: Operating System :: OS Independent

Classifier: Programming Language :: Python

Requires-Python: >=3.7

Description-Content-Type: text/x-rst

License-File: LICENSE.rst

\$ click_

==========

Click is a Python package for creating beautiful command line interfaces in a composable way with as little code as necessary. It's the "Command Line Interface Creation Kit". It's highly configurable but comes with sensible defaults out of the box.

It aims to make the process of writing command line tools quick and fun while also preventing any frustration caused by the inability to implement an intended CLI API.

Click in three points:

- Arbitrary nesting of commands
- Automatic help page generation
- Supports lazy loading of subcommands at runtime

Installing

Install and update using `pip`_:

.. code-block:: text

    $ pip install -U click

.. _pip: https://pip.pypa.io/en/stable/getting-started/

A Simple Example

.. code-block:: python

    import click

    @click.command()
    @click.option("--count", default=1, help="Number of greetings.")
    @click.option("--name", prompt="Your name", help="The person to greet.")
    def hello(count, name):
        """Simple program that greets NAME for a total of COUNT times."""
        for _ in range(count):
            click.echo(f"Hello, {name}!")

    if __name__ == '__main__':
        hello()

.. code-block:: text

    $ python hello.py --count=3
    Your name: Click
    Hello, Click!
    Hello, Click!
    Hello, Click!

Donate

The Pallets organization develops and supports Click and other popular
packages. In order to grow the community of contributors and users, and
allow the maintainers to devote more time to the projects, `please
donate today`_.

.. _please donate today: https://palletsprojects.com/donate

Links

- Documentation: https://click.palletsprojects.com/
- Changes: https://click.palletsprojects.com/changes/
- PyPI Releases: https://pypi.org/project/click/
- Source Code: https://github.com/pallets/click
- Issue Tracker: https://github.com/pallets/click/issues
- Website: https://palletsprojects.com/p/click
- Twitter: https://twitter.com/PalletsTeam
- Chat: https://discord.gg/pallets

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/PKG-INFO
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/src/click.egg-info/PKG-INFO
No license file was found, but licenses were detected in source scan.

Copyright 2001-2006 Gregory P. Ward. All rights reserved.
Copyright 2002-2006 Python Software Foundation. All rights reserved.

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/src/click/parser.py
No license file was found, but licenses were detected in source scan.

Copyright 2014 Pallets

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
    notice, this list of conditions and the following disclaimer in the
    documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/LICENSE.rst
No license file was found, but licenses were detected in source scan.

Including attribution to #612

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/tests/test_formatting.py
No license file was found, but licenses were detected in source scan.

```
[metadata]
name = click
version = attr: click.__version__
url = https://palletsprojects.com/p/click/
project_urls =
 Donate = https://palletsprojects.com/donate
 Documentation = https://click.palletsprojects.com/
 Changes = https://click.palletsprojects.com/changes/
 Source Code = https://github.com/pallets/click/
 Issue Tracker = https://github.com/pallets/click/issues/
 Twitter = https://twitter.com/PalletsTeam
 Chat = https://discord.gg/pallets
license = BSD-3-Clause
license_files = LICENSE.rst
author = Armin Ronacher
author_email = armin.ronacher@active-4.com
maintainer = Pallets
maintainer_email = contact@palletsprojects.com
description = Composable command line interface toolkit
long_description = file: README.rst
long_description_content_type = text/x-rst
classifiers =
 Development Status :: 5 - Production/Stable
 Intended Audience :: Developers
 License :: OSI Approved :: BSD License
 Operating System :: OS Independent
 Programming Language :: Python

[options]
packages = find:
```

```
package_dir = = src
include_package_data = True
python_requires = >= 3.7

[options.packages.find]
where = src

[tool:pytest]
testpaths = tests
filterwarnings =
 error

[coverage:run]
branch = True
source =
 click
 tests

[coverage:paths]
source =
 src
 */site-packages

[flake8]
select = B, E, F, W, B9, ISC
ignore =
 E203
 E501
 E722
 W503
max-line-length = 80
per-file-ignores =
 src/click/__init__.py: F401

[mypy]
files = src/click
python_version = 3.7
show_error_codes = True
disallow_subclassing_any = True
disallow_untyped_calls = True
disallow_untyped_defs = True
disallow_incomplete_defs = True
check_untyped_defs = True
no_implicit_optional = True
local_partial_types = True
no_implicit_reexport = True
strict_equality = True
warn_redundant_casts = True
```

warn_unused_configs = True
warn_unused_ignores = True
warn_return_any = True
warn_unreachable = True

[mypy-colorama.*]
ignore_missing_imports = True

[mypy-importlib_metadata.*]
ignore_missing_imports = True

[egg_info]
tag_build =
tag_date = 0

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/setup.cfg
No license file was found, but licenses were detected in source scan.

.. _contrib:

=============
click-contrib
=============

As the userbase of Click grows, more and more major feature requests pop up in
Click's bugtracker. As reasonable as it may be for those features to be bundled
with Click instead of being a standalone project, many of those requested
features are either highly experimental or have unproven practical use, while
potentially being a burden to maintain.

This is why click-contrib_ exists. The GitHub organization is a collection of
possibly experimental third-party packages whose featureset does not belong
into Click, but also a playground for major features that may be added to Click
in the future. It is also meant to coordinate and concentrate effort on writing
third-party extensions for Click, and to ease the effort of searching for such
extensions. In that sense it could be described as a low-maintenance
alternative to extension repositories of other frameworks.

Please note that the quality and stability of those packages may be different
than what you expect from Click itself. While published under a common
organization, they are still projects separate from Click.

.. _click-contrib: https://github.com/click-contrib/

Found in path(s):
* /opt/cola/permits/1348383324_1656597094.423936/0/click-8-1-3-tar/click-8.1.3/docs/contrib.rst

1.143 python-certifi 2022.12.7

1.143.1 Available under license :

This package contains a modified version of ca-bundle.crt:

ca-bundle.crt -- Bundle of CA Root Certificates

Certificate data from Mozilla as of: Thu Nov 3 19:04:19 2011#
This is a bundle of X.509 certificates of public Certificate Authorities
(CA). These were automatically extracted from Mozilla's root certificates
file (certdata.txt). This file can be found in the mozilla source tree:
https://hg.mozilla.org/mozilla-central/file/tip/security/nss/lib/ckfw/builtins/certdata.txt
It contains the certificates in PEM format and therefore
can be directly used with curl / libcurl / php_curl, or with
an Apache+mod_ssl webserver for SSL client authentication.
Just configure this file as the SSLCACertificateFile.#

***** BEGIN LICENSE BLOCK *****
This Source Code Form is subject to the terms of the Mozilla Public License,
v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain
one at http://mozilla.org/MPL/2.0/.

***** END LICENSE BLOCK *****
@(#) $RCSfile: certdata.txt,v $ $Revision: 1.80 $ $Date: 2011/11/03 15:11:58 $

1.144 gitpython 3.1.31

1.144.1 Available under license :

Copyright (C) 2008, 2009 Michael Trier and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the GitPython project nor the names of
its contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

1.145 platformdirs 3.1.1

1.145.1 Available under license :

MIT License

Copyright (c) 2010-202x The platformdirs developers

1.146 websocket-client 1.5.1

1.146.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2022 engn33r

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

1.147 types-pytz 2022.7.1.2

1.147.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1

Name: types-pytz

Version: 2022.7.1.2

Summary: Typing stubs for pytz

Home-page: https://github.com/python/typeshed

License: Apache-2.0 license

Project-URL: GitHub, https://github.com/python/typeshed

Project-URL: Changes, https://github.com/typeshed-internal/stub_uploader/blob/main/data/changelogs/pytz.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for pytz

This is a PEP 561 type stub package for the `pytz` package. It

can be used by type-checking tools like

[mypy](https://github.com/python/mypy/),

[pyright](https://github.com/microsoft/pyright),

[pytype](https://github.com/google/pytype/),

PyCharm, etc. to check code that uses

`pytz`. The source for this package can be found at

https://github.com/python/typeshed/tree/main/stubs/pytz. All fixes for

types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.

This package was generated from typeshed commit `078c6a095892e12c94462889fa43c2c2dc8280ad`.

Found in path(s):

* /opt/cola/permits/1603644281_1679936793.593923/0/types-pytz-2022-7-1-2-tar-gz/types-pytz-
2022.7.1.2/types_pytz.egg-info/PKG-INFO

* /opt/cola/permits/1603644281_1679936793.593923/0/types-pytz-2022-7-1-2-tar-gz/types-pytz-2022.7.1.2/PKG-
INFO

1.148 isort 5.12.0

1.148.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 Timothy Edmund Crosley

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
MIT License

Copyright (c) 2021 Taneli Hukkinen

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.149 bullet 2.2.0

1.149.1 Available under license :

Copyright (c) 2009 - 2010 Richard Huang (flyerhzm@gmail.com)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.150 py-filelock 3.10.0

1.150.1 Available under license :

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or
distribute this software, either in source code form or as a compiled
binary, for any purpose, commercial or non-commercial, and by any
means.

In jurisdictions that recognize copyright laws, the author or authors
of this software dedicate any and all copyright interest in the
software to the public domain. We make this dedication for the benefit
of the public at large and to the detriment of our heirs and
successors. We intend this dedication to be an overt act of
relinquishment in perpetuity of all present and future rights to this
software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org>

1.151 orjson 3.8.7

1.151.1 Available under license :

Permission is hereby granted, free of charge, to any
person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the
Software without restriction, including without
limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software
is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice
shall be included in all copies or substantial portions
of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED
TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT
SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR
IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

                    Apache License
              Version 2.0, January 2004
             http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

  "License" shall mean the terms and conditions for use, reproduction,
  and distribution as defined by Sections 1 through 9 of this document.

  "Licensor" shall mean the copyright owner or entity authorized by
  the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

 http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.152 python-six 1.16.0

1.152.1 Available under license :

The primary author and maintainer of six is Benjamin Peterson. He would like to acknowledge the following people who submitted bug reports, pull requests, and otherwise worked to improve six:

Marc Abramowitz

immerrr again

Alexander Artemenko

Aymeric Augustin

Lee Ball

Ben Bariteau

Ned Batchelder

Wouter Bolsterlee

Brett Cannon

Jason R. Coombs

Julien Danjou

Ben Darnell

Ben Davis

Jon Dufresne

Tim Graham

Thomas Grainger

Max Grender-Jones

Pierre Grimaud

Joshua Harlow

Toshiki Kataoka

Hugo van Kemenade

Anselm Kruis

Ivan Levkivskyi

Alexander Lukanin

James Mills

Jordan Moldow

Berker Peksag

Sridhar Ratnakumar

Erik Rose

Mirko Rossini

Peter Ruibal

Miroslav Shubernetskiy

Eli Schwartz

Anthony Sottile

Victor Stinner

Jonathan Vanasco

Lucas Wiman

Jingxin Zhu

If you think you belong on this list, please let me know! --Benjamin

this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.153 pytz-python-time-zone-library 2022.7.1

1.153.1 Available under license :

Copyright (c) 2003-2019 Stuart Bishop <stuart@stuartbishop.net>

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

1.154 pprintpp 0.4.0

1.154.1 Available under license :

tl;dr: all code code is licensed under simplified BSD, unless stated otherwise.

Unless stated otherwise in the source files, all code is copyright 2010 David
Wolever <david@wolever.net>. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY <COPYRIGHT HOLDER> ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL <COPYRIGHT HOLDER> OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of David Wolever.

1.155 typer 0.7.0

1.155.1 Available under license :

The MIT License (MIT)

Copyright (c) 2019 Sebastin Ramrez

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.156 importlib-resources 5.12.0

1.156.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.157 httpx 0.23.3

1.157.1 Available under license :

Copyright 2019, [Encode OSS Ltd](https://www.encode.io/).
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.158 python-semver 2.13.0

1.158.1 Available under license :

############
Contributors
############

Python SemVer library
#####################

This document records the primary maintainers and significant
contributors to this code base.

Thank you to everyone whose work has made this possible.

Primary maintainers
===================

* Kostiantyn Rybnikov <k-bx@k-bx.com>

* Sbastien Celles <s.celles@gmail.com>

Significant contributors

========================

* Alexander Puzynia <werwolf.by@gmail.com>

* Alexander Shorin <kxepal@gmail.com>

* Anton Talevnin <TalAntR@users.noreply.github.com>

* Ben Finney <ben+python@benfinney.id.au>

* Carles Barrobes <carles@barrobes.com>

* Craig Blaszczyk <masterjakul@gmail.com>

* Damien Nad <anvil@users.noreply.github.com>

* Eli Bishop <eli-darkly@users.noreply.github.com>

* George Sakkis <gsakkis@users.noreply.github.com>

* Jan Pieter Waagmeester <jieter@jieter.nl>

* Jelo Agnasin <jelo@icannhas.com>

* Karol Werner <karol.werner@ppkt.eu>

* Peter Bittner <django@bittner.it>

* robi-wan <robi-wan@suyu.de>

* sbrudenell <sbrudenell@users.noreply.github.com>

* T. Jameson Little <t.jameson.little@gmail.com>

* Tom Schraitle <tom_schr@web.de>

* Thomas Laferriere <tlaferriere@users.noreply.github.com>

* Tuure Laurinolli <tuure@laurinolli.net>

* Tyler Cross <tyler@crosscollab.com>

* Zack Lalanne <zack.lalanne@gmail.com>

..

  Local variables:

  coding: utf-8

  mode: text

  mode: rst

  End:

  vim: fileencoding=utf-8 filetype=rst :

1.159 colorama 0.4.6

1.159.1 Available under license :

list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors
 may be used to endorse or promote products derived from this software without
 specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.160 tomli 2.0.1

1.160.1 Available under license :

MIT License

Copyright (c) 2021 Taneli Hukkinen

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.161 flake8-bugbear 23.3.12

1.161.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016 ukasz Langa

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.162 flake8-fixme 1.1.1

1.162.1 Available under license :

No license file was found, but licenses were detected in source scan.

License: Apache-2.0

Found in path(s):
* /opt/cola/permits/1343905265_1655169447.5720215/0/flake8-fixme-1-1-1-tar-gz/flake8-fixme-1.1.1/PKG-INFO
* /opt/cola/permits/1343905265_1655169447.5720215/0/flake8-fixme-1-1-1-tar-gz/flake8-fixme-
1.1.1/flake8_fixme.egg-info/PKG-INFO

1.163 flakehell 0.9.0

1.163.1 Available under license :

MIT License 2019 Gram (@orsinium) <master_fess@mail.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.164 requests-oauthlib 1.3.1
1.164.1 Available under license :

ISC License

Copyright (c) 2014 Kenneth Reitz.

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.165 types-tabulate 0.9.0.1
1.165.1 Available under license :

No license file was found, but licenses were detected in source scan.

Metadata-Version: 2.1
Name: types-tabulate
Version: 0.9.0.1
Summary: Typing stubs for tabulate
Home-page: https://github.com/python/typeshed
License: Apache-2.0 license
Project-URL: GitHub, https://github.com/python/typeshed

Project-URL: Changes, https://github.com/typeshed-internal/stub_uploader/blob/main/data/changelogs/tabulate.md

Project-URL: Issue tracker, https://github.com/python/typeshed/issues

Project-URL: Chat, https://gitter.im/python/typing

Classifier: License :: OSI Approved :: Apache Software License

Classifier: Programming Language :: Python :: 3

Classifier: Typing :: Stubs Only

Description-Content-Type: text/markdown

Typing stubs for tabulate

This is a PEP 561 type stub package for the `tabulate` package. It
can be used by type-checking tools like
[mypy](https://github.com/python/mypy/),
[pyright](https://github.com/microsoft/pyright),
[pytype](https://github.com/google/pytype/),
PyCharm, etc. to check code that uses
`tabulate`. The source for this package can be found at
https://github.com/python/typeshed/tree/main/stubs/tabulate. All fixes for
types and metadata should be contributed there.

See https://github.com/python/typeshed/blob/main/README.md for more details.
This package was generated from typeshed commit `880c0da4045cd5ff2c29b73429629adf27e49d50`.

Found in path(s):
* /opt/cola/permits/1603644200_1679935657.008714/0/types-tabulate-0-9-0-1-tar-gz/types-tabulate-0.9.0.1/PKG-INFO
* /opt/cola/permits/1603644200_1679935657.008714/0/types-tabulate-0-9-0-1-tar-gz/types-tabulate-0.9.0.1/types_tabulate.egg-info/PKG-INFO

1.166 flake8-docstrings 1.7.0

1.166.1 Available under license :

Copyright (c) 2013 Simon Andre, Ian Cordasco

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.167 pathspec 0.11.1

1.167.1 Available under license :

Mozilla Public License Version 2.0
==================================

1. Definitions

1.1. "Contributor"
   means each individual or legal entity that creates, contributes to
   the creation of, or owns Covered Software.

1.2. "Contributor Version"
   means the combination of the Contributions of others (if any) used
   by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
   means Covered Software of a particular Contributor.

1.4. "Covered Software"
   means Source Code Form to which the initial Contributor has attached
   the notice in Exhibit A, the Executable Form of such Source Code
   Form, and Modifications of such Source Code Form, in each case
   including portions thereof.

1.5. "Incompatible With Secondary Licenses"
   means

   (a) that the initial Contributor has attached the notice described
       in Exhibit B to the Covered Software; or

   (b) that the Covered Software was made available under the terms of
       version 1.1 or earlier of the License, but not also under the
       terms of a Secondary License.

1.6. "Executable Form"
   means any form of the work other than Source Code Form.

1.7. "Larger Work"
   means a work that combines Covered Software with other material, in
   a separate file or files, that is not Covered Software.

1.8. "License"
  means this document.

1.9. "Licensable"
  means having the right to grant, to the maximum extent possible,
  whether at the time of the initial grant or subsequently, any and
  all of the rights conveyed by this License.

1.10. "Modifications"
  means any of the following:

  (a) any file in Source Code Form that results from an addition to,
      deletion from, or modification of the contents of Covered
      Software; or

  (b) any new file in Source Code Form that contains any Covered
      Software.

1.11. "Patent Claims" of a Contributor
  means any patent claim(s), including without limitation, method,
  process, and apparatus claims, in any patent Licensable by such
  Contributor that would be infringed, but for the grant of the
  License, by the making, using, selling, offering for sale, having
  made, import, or transfer of either its Contributions or its
  Contributor Version.

1.12. "Secondary License"
  means either the GNU General Public License, Version 2.0, the GNU
  Lesser General Public License, Version 2.1, the GNU Affero General
  Public License, Version 3.0, or any later versions of those
  licenses.

1.13. "Source Code Form"
  means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
  means an individual or a legal entity exercising rights under this
  License. For legal entities, "You" includes any entity that
  controls, is controlled by, or is under common control with You. For
  purposes of this definition, "control" means (a) the power, direct
  or indirect, to cause the direction or management of such entity,
  whether by contract or otherwise, or (b) ownership of more than
  fifty percent (50%) of the outstanding shares or beneficial
  ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to

distribute the Covered Software under a subsequent version of this
License (see Section 10.2) or under the terms of a Secondary License (if
permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its
Contributions are its original creation(s) or it has sufficient rights
to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under
applicable copyright doctrines of fair use, fair dealing, or other
equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted
in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any
Modifications that You create or to which You contribute, must be under
the terms of this License. You must inform recipients that the Source
Code Form of the Covered Software is governed by the terms of this
License, and how they can obtain a copy of this License. You may not
attempt to alter or restrict the recipients' rights in the Source Code
Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code
    Form, as described in Section 3.1, and You must inform recipients of
    the Executable Form how they can obtain a copy of such Source Code
    Form by reasonable means in a timely manner, at a charge no more
    than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this
    License, or sublicense it under different terms, provided that the
    license for the Executable Form does not attempt to limit or alter
    the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

```
************************************************************************
*                                                                    *
*  6. Disclaimer of Warranty                                         *
*  -------------------------                                         *
*                                                                    *
*  Covered Software is provided under this License on an "as is"     *
*  basis, without warranty of any kind, either expressed, implied, or *
*  statutory, including, without limitation, warranties that the     *
*  Covered Software is free of defects, merchantable, fit for a      *
*  particular purpose or non-infringing. The entire risk as to the   *
*  quality and performance of the Covered Software is with You.      *
*  Should any Covered Software prove defective in any respect, You    *
*  (not any Contributor) assume the cost of any necessary servicing,  *
*  repair, or correction. This disclaimer of warranty constitutes an  *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.       *
*                                                                    *
************************************************************************
```

```
**********************************************************************
*                                      *
* 7. Limitation of Liability                *
* -------------------------                *
*                                      *
* Under no circumstances and under no legal theory, whether tort  *
* (including negligence), contract, or otherwise, shall any      *
* Contributor, or anyone who distributes Covered Software as     *
* permitted above, be liable to You for any direct, indirect,    *
* special, incidental, or consequential damages of any character  *
* including, without limitation, damages for lost profits, loss of  *
* goodwill, work stoppage, computer failure or malfunction, or any  *
* and all other commercial damages or losses, even if such party   *
* shall have been informed of the possibility of such damages. This *
* limitation of liability shall not apply to liability for death or  *
* personal injury resulting from such party's negligence to the    *
* extent applicable law prohibits such limitation. Some        *
* jurisdictions do not allow the exclusion or limitation of      *
* incidental or consequential damages, so this exclusion and      *
* limitation may not apply to You.                 *
*                                      *
**********************************************************************
```

8. Litigation

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

  This Source Code Form is subject to the terms of the Mozilla Public
  License, v. 2.0. If a copy of the MPL was not distributed with this
  file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

  This Source Code Form is "Incompatible With Secondary Licenses", as
  defined by the Mozilla Public License, v. 2.0.

1.168 flake8-eradicate 1.4.0

1.168.1 Available under license :

MIT License

Copyright (c) 2018 Nikita Sobolev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.169 kubetest 0.9.3

1.169.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

 The GNU General Public License is a free, copyleft license for
software and other kinds of works.

 The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to

any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run
modified versions of the software inside them, although the manufacturer
can do so. This is fundamentally incompatible with the aim of
protecting users' freedom to change the software. The systematic
pattern of such abuse occurs in the area of products for individuals to
use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those
products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions
of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.
States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could
make it effectively proprietary. To prevent this, the GPL assures that
patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work

for making modifications to it. "Object code" means any non-source form of a work.

 A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

 The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

 The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

 The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

 The Corresponding Source for a work in source code form is that same work.

 2. Basic Permissions.

 All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to

produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

  a) The work must carry prominent notices stating that you modified
  it, and giving a relevant date.

  b) The work must carry prominent notices stating that it is
  released under this License and any conditions added under section
  7. This requirement modifies the requirement in section 4 to
  "keep intact all notices".

  c) You must license the entire work, as a whole, under this
  License to anyone who comes into possession of a copy. This
  License will therefore apply, along with any applicable section 7
  additional terms, to the whole of the work, and all its parts,
  regardless of how they are packaged. This License gives no
  permission to license the work in any other way, but it does not
  invalidate such permission if you have separately received it.

  d) If the work has interactive user interfaces, each must display
  Appropriate Legal Notices; however, if the Program has interactive
  interfaces that do not display Appropriate Legal Notices, your
  work need not make them do so.

 A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

 6. Conveying Non-Source Forms.

 You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

  a) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by the
  Corresponding Source fixed on a durable physical medium
  customarily used for software interchange.

  b) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by a

written offer, valid for at least three years and valid for as
long as you offer spare parts or customer support for that product
model, to give anyone who possesses the object code either (1) a
copy of the Corresponding Source for all the software in the
product that is covered by this License, on a durable physical
medium customarily used for software interchange, for a price no
more than your reasonable cost of physically performing this
conveying of source, or (2) access to copy the
Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the
written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord
with subsection 6b.

d) Convey the object code by offering access from a designated
place (gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party)
that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the
Corresponding Source. Regardless of what server hosts the
Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided
you inform other peers where the object code and Corresponding
Source of the work are being offered to the general public at no
charge under subsection 6d.

 A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.

 A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family,
or household purposes, or (2) anything designed or sold for incorporation
into a dwelling. In determining whether a product is a consumer product,
doubtful cases shall be resolved in favor of coverage. For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product. A product
is a consumer product regardless of whether the product has substantial

commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source. The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.

If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as
part of a transaction in which the right of possession and use of the
User Product is transferred to the recipient in perpetuity or for a
fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied
by the Installation Information. But this requirement does not apply
if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has
been installed in ROM).

The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates
for a work that has been modified or installed by the recipient, or for
the User Product in which it has been modified or installed. Access to a
network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law. If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option

remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own
removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

 Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of
that material) supplement the terms of this License with terms:

  a) Disclaiming warranty or limiting liability differently from the
  terms of sections 15 and 16 of this License; or

  b) Requiring preservation of specified reasonable legal notices or
  author attributions in that material or in the Appropriate Legal
  Notices displayed by works containing it; or

  c) Prohibiting misrepresentation of the origin of that material, or
  requiring that modified versions of such material be marked in
  reasonable ways as different from the original version; or

  d) Limiting the use for publicity purposes of names of licensors or
  authors of the material; or

  e) Declining to grant rights under trademark law for use of some
  trade names, trademarks, or service marks; or

  f) Requiring indemnification of licensors and authors of that
  material by anyone who conveys the material (or modified versions of
  it) with contractual assumptions of liability to the recipient, for
  any liability that these contractual assumptions directly impose on
  those licensors and authors.

 All other non-permissive additional terms are considered "further
restrictions" within the meaning of section 10. If the Program as you
received it, or any part of it, contains a notice stating that it is
governed by this License along with a term that is a further
restriction, you may remove that term. If a license document contains
a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms
of that license document, provided that the further restriction does
not survive such relicensing or conveying.

 If you add terms to a covered work in accord with this section, you
must place, in the relevant source files, a statement of the
additional terms that apply to those files, or a notice indicating
where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible

for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may

not convey it at all. For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

  13. Use with the GNU Affero General Public License.

 Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

  14. Revised Versions of this License.

 The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

 Each version is given a distinguishing version number. If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published
by the Free Software Foundation.

 If the Program specifies that a proxy can decide which future
versions of the GNU General Public License can be used, that proxy's
public statement of acceptance of a version permanently authorizes you
to choose that version for the Program.

 Later license versions may give you additional or different
permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a
later version.

  15. Disclaimer of Warranty.

 THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,

THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

  16. Limitation of Liability.

  IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

  17. Interpretation of Sections 15 and 16.

  If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

                    END OF TERMS AND CONDITIONS

            How to Apply These Terms to Your New Programs

  If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

  To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year> <name of author>

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

 If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

 You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

 The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.170 jmespath-py 1.0.1

1.170.1 Available under license :

Copyright (c) 2013 Amazon.com, Inc. or its affiliates. All Rights Reserved

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, dis-
tribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the fol-
lowing conditions:

The above copyright notice and this permission notice shall be included

in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABIL-
ITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT
SHALL THE AUTHOR BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.171 idna 3.4

1.171.1 Available under license :

BSD 3-Clause License

Copyright (c) 2013-2021, Kim Davies
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.172 black 23.1.0

1.172.1 Available under license :

Frequently Asked Questions

The most common questions and issues users face are aggregated to this FAQ.

```{contents}
:local:
:backlinks: none
:class: this-will-duplicate-information-and-it-is-still-useful-here
```

Why spaces? I prefer tabs

PEP 8 recommends spaces over tabs, and they are used by most of the Python community.
Black provides no options to configure the indentation style, and requests for such
options will not be considered.

However, we recognise that using tabs is an accessibility issue as well. While the
option will never be added to _Black_, visually impaired developers may find conversion
tools such as `expand/unexpand` (for Linux) useful when contributing to Python projects.
A workflow might consist of e.g. setting up appropriate pre-commit and post-merge git
hooks, and scripting `unexpand` to run after applying _Black_.

Does Black have an API?

Not yet. _Black_ is fundamentally a command line tool. Many
[integrations](integrations/index.rst) are provided, but a Python interface is not one
of them. A simple API is being [planned](https://github.com/psf/black/issues/779)
though.

Is Black safe to use?

Yes. _Black_ is strictly about formatting, nothing else. Black strives to ensure that
after formatting the AST is
[checked](the_black_code_style/current_style.md#ast-before-and-after-formatting) with
limited special cases where the code is allowed to differ. If issues are found, an error
is raised and the file is left untouched. Magical comments that influence linters and
other tools, such as `# noqa`, may be moved by _Black_. See below for more details.

How stable is Black's style?

Stable. _Black_ aims to enforce one style and one style only, with some room for
pragmatism. See [The Black Code Style](the_black_code_style/index.rst) for more details.

Starting in 2022, the formatting output will be stable for the releases made in the same
year (other than unintentional bugs). It is possible to opt-in to the latest formatting
styles, using the `--preview` flag.

Why is my file not formatted?

Most likely because it is ignored in `.gitignore` or excluded with configuration. See
[file collection and discovery](usage_and_configuration/file_collection_and_discovery.md)
for details.

Why is my Jupyter Notebook cell not formatted?

Black is timid about formatting Jupyter Notebooks. Cells containing any of the
following will not be formatted:

- automagics (e.g. `pip install black`)
- non-Python cell magics (e.g. `%%writeline`). These can be added with the flag
`--python-cell-magics`, e.g. `black --python-cell-magics writeline hello.ipynb`.
- multiline magics, e.g.:

```python
%timeit f(1, \
    2, \
    3)
```

- code which `IPython`'s `TransformerManager` would transform magics into, e.g.:

```python
get_ipython().system('ls')
```

- invalid syntax, as it can't be safely distinguished from automagics in the absence of
a running `IPython` kernel.

Why are Flake8's E203 and W503 violated?

Because they go against PEP 8. E203 falsely triggers on list
[slices](the_black_code_style/current_style.md#slices), and adhering to W503 hinders
readability because operators are misaligned. Disable W503 and enable the
disabled-by-default counterpart W504. E203 should be disabled while changes are still
[discussed](https://github.com/PyCQA/pycodestyle/issues/373).

Which Python versions does Black support?

Currently the runtime requires Python 3.7-3.11. Formatting is supported for files
containing syntax from Python 3.3 to 3.11. We promise to support at least all Python
versions that have not reached their end of life. This is the case for both running
Black and formatting code.

Support for formatting Python 2 code was removed in version 22.0. While we've made no
plans to stop supporting older Python 3 minor versions immediately, their support might

also be removed some time in the future without a deprecation period.

Runtime support for 3.6 was removed in version 22.10.0.

Why does my linter or typechecker complain after I format my code?

Some linters and other tools use magical comments (e.g., `# noqa`, `# type: ignore`) to influence their behavior. While Black does its best to recognize such comments and leave them in the right place, this detection is not and cannot be perfect. Therefore, you'll sometimes have to manually move these comments to the right place after you format your codebase with _Black_.

Can I run Black with PyPy?

Yes, there is support for PyPy 3.7 and higher.

Why does Black not detect syntax errors in my code?

Black is an autoformatter, not a Python linter or interpreter. Detecting all syntax errors is not a goal. It can format all code accepted by CPython (if you find an example where that doesn't hold, please report a bug!), but it may also format some code that CPython doesn't accept.

(labels/mypyc-support)=

What is `compiled: yes/no` all about in the version output?

While _Black_ is indeed a pure Python project, we use [mypyc] to compile _Black_ into a C Python extension, usually doubling performance. These compiled wheels are available for 64-bit versions of Windows, Linux (via the manylinux standard), and macOS across all supported CPython versions.

Platforms including musl-based and/or ARM Linux distributions, and ARM Windows are currently **not** supported. These platforms will fall back to the slower pure Python wheel available on PyPI.

If you are experiencing exceptionally weird issues or even segfaults, you can try passing `--no-binary black` to your pip install invocation. This flag excludes all wheels (including the pure Python wheel), so this command will use the [sdist].

[mypyc]: https://mypyc.readthedocs.io/en/latest/
[sdist]:
 https://packaging.python.org/en/latest/glossary/#term-Source-Distribution-or-sdist
[report]
omit =
 src/blib2to3/*
 tests/data/*
 /site-packages/

```
    .tox/*

[run]
relative_files = True
---
name: Style issue
about: Help us improve the Black style
title: ""
labels: "T: design"
assignees: ""
---

**Describe the style change**

<!-- A clear and concise description of how the style can be
improved. -->

**Examples in the current _Black_ style**

<!-- Think of some short code snippets that show
how the current _Black_ style is not great: -->

```python
def f():
    "Make sure this code is blackened"""
    pass
```

Desired style

<!-- How do you think _Black_ should format the above snippets: -->

```python
def f(
    ):
    pass
```

Additional context

<!-- Add any other context about the problem here. -->
foo
- id: black
  name: black
  description: "Black: The uncompromising Python code formatter"
  entry: black
  language: python
  minimum_pre_commit_version: 2.9.2
```

```
require_serial: true
types_or: [python, pyi]
- id: black-jupyter
name: black-jupyter
description:
  "Black: The uncompromising Python code formatter (with Jupyter Notebook support)"
entry: black
language: python
minimum_pre_commit_version: 2.9.2
require_serial: true
types_or: [python, pyi, jupyter]
additional_dependencies: [".[jupyter]"]
```

A. HISTORY OF THE SOFTWARE
==========================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see https://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see https://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations, which became
Zope Corporation. In 2001, the Python Software Foundation (PSF, see
https://www.python.org/psf/) was formed, a non-profit organization
created specifically to own Python-related Intellectual Property.
Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see https://opensource.org for
the Open Source Definition). Historically, most, but not all, Python
releases have also been GPL-compatible; the table below summarizes
the various releases.

Release	Derived from	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes
1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes
1.6	1.5.2	2000	CNRI	no
2.0	1.6	2000	BeOpen.com	no
1.6.1	1.6	2001	CNRI	yes (2)
2.1	2.0+1.6.1	2001	PSF	no
2.0.1	2.0+1.6.1	2001	PSF	yes

```
2.1.1       2.1+2.0.1   2001      PSF        yes
2.1.2       2.1.1      2002     PSF       yes
2.1.3       2.1.2      2002     PSF       yes
2.2 and above  2.1.1      2001-now   PSF       yes
```

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under
    the GPL. All Python licenses, unlike the GPL, let you distribute
    a modified version without making your changes open source. The
    GPL-compatible licenses make it possible to combine Python with
    other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible,
    because its license has a choice of law clause. According to
    CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1
    is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's
direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
===

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
----

1. This LICENSE AGREEMENT is between the Python Software Foundation
("PSF"), and the Individual or Organization ("Licensee") accessing and
otherwise using this software ("Python") in source or binary form and
its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby
grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works,
distribute, and otherwise use Python alone or in any derivative version,
provided, however, that PSF's License Agreement and PSF's notice of copyright,
i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018 Python Software Foundation; All
Rights Reserved" are retained in Python alone or in any derivative version
prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR

IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE
SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS
AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY
DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all
respects by the law of the State of California, excluding conflict of
law provisions. Nothing in this License Agreement shall be deemed to
create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant
permission to use BeOpen trademarks or trade names in a trademark
sense to endorse or promote products or services of Licensee, or any
third party. As an exception, the "BeOpen Python" logos available at
http://www.pythonlabs.com/logos.html may be used according to the
permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
----

1. This LICENSE AGREEMENT is between the Corporation for National
Research Initiatives, having an office at 1895 Preston White Drive,
Reston, VA 20191 ("CNRI"), and the Individual or Organization
("Licensee") accessing and otherwise using Python 1.6.1 software in
source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a nonexclusive, royalty-free, world-wide
license to reproduce, analyze, test, perform and/or display publicly,
prepare derivative works, distribute, and otherwise use Python 1.6.1
alone or in any derivative version, provided, however, that CNRI's
License Agreement and CNRI's notice of copyright, i.e., "Copyright (c)
1995-2001 Corporation for National Research Initiatives; All Rights
Reserved" are retained in Python 1.6.1 alone or in any derivative
version prepared by Licensee. Alternately, in lieu of CNRI's License
Agreement, Licensee may substitute the following text (omitting the

quotes): "Python 1.6.1 is made available subject to the terms and
conditions in CNRI's License Agreement. This Agreement together with
Python 1.6.1 may be located on the Internet using the following
unique, persistent identifier (known as a handle): 1895.22/1013. This
Agreement may also be obtained from a proxy server on the Internet
using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python 1.6.1 or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS"
basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. This License Agreement shall be governed by the federal
intellectual property law of the United States, including without
limitation the federal copyright law, and, to the extent such
U.S. federal law does not apply, by the law of the Commonwealth of
Virginia, excluding Virginia's conflict of law provisions.
Notwithstanding the foregoing, with regard to derivative works based
on Python 1.6.1 that incorporate non-separable material that was
previously distributed under the GNU General Public License (GPL), the
law of the Commonwealth of Virginia shall govern this License
Agreement only as to issues arising under or with respect to
Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this
License Agreement shall be deemed to create any relationship of
agency, partnership, or joint venture between CNRI and Licensee. This
License Agreement does not grant permission to use CNRI trademarks or
trade name in a trademark sense to endorse or promote products or
services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying,
installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Authors

Glued together by [ukasz Langa](mailto:lukasz@langa.pl).

Maintained with:

- [Carol Willing](mailto:carolcode@willingconsulting.com)
- [Carl Meyer](mailto:carl@oddbird.net)
- [Jelle Zijlstra](mailto:jelle.zijlstra@gmail.com)
- [Mika Naylor](mailto:mail@autophagy.io)
- [Zsolt Dollenstein](mailto:zsol.zsol@gmail.com)
- [Cooper Lees](mailto:me@cooperlees.com)
- [Richard Si](mailto:sichard26@gmail.com)
- [Felix Hildn](mailto:felix.hilden@gmail.com)
- [Batuhan Taskaya](mailto:batuhan@python.org)

Multiple contributions by:

- [Abdur-Rahmaan Janhangeer](mailto:arj.python@gmail.com)
- [Adam Johnson](mailto:me@adamj.eu)
- [Adam Williamson](mailto:adamw@happyassassin.net)
- [Alexander Huynh](mailto:ahrex-gh-psf-black@e.sc)

- [Alexandr Artemyev](mailto:mogost@gmail.com)
- [Alex Vandiver](mailto:github@chmrr.net)
- [Allan Simon](mailto:allan.simon@supinfo.com)
- Anders-Petter Ljungquist
- [Amethyst Reese](mailto:amy@n7.gg)
- [Andrew Thorp](mailto:andrew.thorp.dev@gmail.com)
- [Andrew Zhou](mailto:andrewfzhou@gmail.com)
- [Andrey](mailto:dyuuus@yandex.ru)
- [Andy Freeland](mailto:andy@andyfreeland.net)
- [Anthony Sottile](mailto:asottile@umich.edu)
- [Antonio Ossa Guerra](mailto:aaossa+black@uc.cl)
- [Arjaan Buijk](mailto:arjaan.buijk@gmail.com)
- [Arnav Borbornah](mailto:arnavborborah11@gmail.com)
- [Artem Malyshev](mailto:proofit404@gmail.com)
- [Asger Hautop Drewsen](mailto:asgerdrewsen@gmail.com)
- [Augie Fackler](mailto:raf@durin42.com)
- [Aviskar KC](mailto:aviskarkc10@gmail.com)
- Batuhan Takaya
- [Benjamin Wohlwend](mailto:bw@piquadrat.ch)
- [Benjamin Woodruff](mailto:github@benjam.info)
- [Bharat Raghunathan](mailto:bharatraghunthan9767@gmail.com)
- [Brandt Bucher](mailto:brandtbucher@gmail.com)
- [Brett Cannon](mailto:brett@python.org)
- [Bryan Bugyi](mailto:bryan.bugyi@rutgers.edu)
- [Bryan Forbes](mailto:bryan@reigndropsfall.net)
- [Calum Lind](mailto:calumlind@gmail.com)
- [Charles](mailto:peacech@gmail.com)
- Charles Reid
- [Christian Clauss](mailto:cclauss@bluewin.ch)
- [Christian Heimes](mailto:christian@python.org)
- [Chuck Wooters](mailto:chuck.wooters@microsoft.com)
- [Chris Rose](mailto:offline@offby1.net)
- Codey Oxley
- [Cong](mailto:congusbongus@gmail.com)
- [Cooper Ry Lees](mailto:me@cooperlees.com)
- [Dan Davison](mailto:dandavison7@gmail.com)
- [Daniel Hahler](mailto:github@thequod.de)
- [Daniel M. Capella](mailto:polycitizen@gmail.com)
- Daniele Esposti
- [David Hotham](mailto:david.hotham@metaswitch.com)
- [David Lukes](mailto:dafydd.lukes@gmail.com)
- [David Szotten](mailto:davidszotten@gmail.com)
- [Denis Laxalde](mailto:denis@laxalde.org)
- [Douglas Thor](mailto:dthor@transphormusa.com)
- dylanjblack
- [Eli Treuherz](mailto:eli@treuherz.com)
- [Emil Hessman](mailto:emil@hessman.se)
- [Felix Kohlgrber](mailto:felix.kohlgrueber@gmail.com)

- [Florent Thiery](mailto:fthiery@gmail.com)
- Francisco
- [Giacomo Tagliabue](mailto:giacomo.tag@gmail.com)
- [Greg Gandenberger](mailto:ggandenberger@shoprunner.com)
- [Gregory P. Smith](mailto:greg@krypto.org)
- Gustavo Camargo
- hauntsaninja
- [Hadi Alqattan](mailto:alqattanhadizaki@gmail.com)
- [Hassan Abouelela](mailto:hassan@hassanamr.com)
- [Heaford](mailto:dan@heaford.com)
- [Hugo Barrera](mailto::hugo@barrera.io)
- Hugo van Kemenade
- [Hynek Schlawack](mailto:hs@ox.cx)
- [Ionite](mailto:dev@ionite.io)
- [Ivan Katani](mailto:ivan.katanic@gmail.com)
- [Jakub Kadlubiec](mailto:jakub.kadlubiec@skyscanner.net)
- [Jakub Warczarek](mailto:jakub.warczarek@gmail.com)
- [Jan Hntek](mailto:jan.hnatek@gmail.com)
- [Jason Fried](mailto:me@jasonfried.info)
- [Jason Friedland](mailto:jason@friedland.id.au)
- [jgirardet](mailto:ijkl@netc.fr)
- Jim Brnnlund
- [Jimmy Jia](mailto:tesrin@gmail.com)
- [Joe Antonakakis](mailto:jma353@cornell.edu)
- [Jon Dufresne](mailto:jon.dufresne@gmail.com)
- [Jonas Obrist](mailto:ojiidotch@gmail.com)
- [Jonty Wareing](mailto:jonty@jonty.co.uk)
- [Jose Nazario](mailto:jose.monkey.org@gmail.com)
- [Joseph Larson](mailto:larson.joseph@gmail.com)
- [Josh Bode](mailto:joshbode@fastmail.com)
- [Josh Holland](mailto:anowlcalledjosh@gmail.com)
- [Joshua Cannon](mailto:joshdcannon@gmail.com)
- [Jos Padilla](mailto:jpadilla@webapplicate.com)
- [Juan Luis Cano Rodrguez](mailto:hello@juanlu.space)
- [kaiix](mailto:kvn.hou@gmail.com)
- [Katie McLaughlin](mailto:katie@glasnt.com)
- Katrin Leinweber
- [Keith Smiley](mailto:keithbsmiley@gmail.com)
- [Kenyon Ralph](mailto:kenyon@kenyonralph.com)
- [Kevin Kirsche](mailto:Kev.Kirsche+GitHub@gmail.com)
- [Kyle Hausmann](mailto:kyle.hausmann@gmail.com)
- [Kyle Sunden](mailto:sunden@wisc.edu)
- Lawrence Chan
- [Linus Groh](mailto:mail@linusgroh.de)
- [Loren Carvalho](mailto:comradeloren@gmail.com)
- [Luka Sterbic](mailto:luka.sterbic@gmail.com)
- [LukasDrude](mailto:mail@lukas-drude.de)
- Mahmoud Hossam

- Mariatta
- [Matt VanEseltine](mailto:vaneseltine@gmail.com)
- [Matthew Clapp](mailto:itsayellow+dev@gmail.com)
- [Matthew Walster](mailto:matthew@walster.org)
- Max Smolens
- [Michael Aquilina](mailto:michaelaquilina@gmail.com)
- [Michael Flaxman](mailto:michael.flaxman@gmail.com)
- [Michael J. Sullivan](mailto:sully@msully.net)
- [Michael McClimon](mailto:michael@mcclimon.org)
- [Miguel Gaiowski](mailto:miggaiowski@gmail.com)
- [Mike](mailto:roshi@fedoraproject.org)
- [mikehoyio](mailto:mikehoy@gmail.com)
- [Min ho Kim](mailto:minho42@gmail.com)
- [Miroslav Shubernetskiy](mailto:miroslav@miki725.com)
- MomIsBestFriend
- [Nathan Goldbaum](mailto:ngoldbau@illinois.edu)
- [Nathan Hunt](mailto:neighthan.hunt@gmail.com)
- [Neraste](mailto:neraste.herr10@gmail.com)
- [Nikolaus Waxweiler](mailto:madigens@gmail.com)
- [Ofek Lev](mailto:ofekmeister@gmail.com)
- [Osaetin Daniel](mailto:osaetindaniel@gmail.com)
- [otstrel](mailto:otstrel@gmail.com)
- [Pablo Galindo](mailto:Pablogsal@gmail.com)
- [Paul Ganssle](mailto:p.ganssle@gmail.com)
- [Paul Meinhardt](mailto:mnhrdt@gmail.com)
- [Peter Bengtsson](mailto:mail@peterbe.com)
- [Peter Grayson](mailto:pete@jpgrayson.net)
- [Peter Stensmyr](mailto:peter.stensmyr@gmail.com)
- pmacosta
- [Quentin Pradet](mailto:quentin@pradet.me)
- [Ralf Schmitt](mailto:ralf@systemexit.de)
- [Ramn Valles](mailto:mroutis@protonmail.com)
- [Richard Fearn](mailto:richardfearn@gmail.com)
- [Rishikesh Jha](mailto:rishijha424@gmail.com)
- [Rupert Bedford](mailto:rupert@rupertb.com)
- Russell Davis
- [Sagi Shadur](mailto:saroad2@gmail.com)
- [Rmi Verschelde](mailto:rverschelde@gmail.com)
- [Sami Salonen](mailto:sakki@iki.fi)
- [Samuel Cormier-Iijima](mailto:samuel@cormier-iijima.com)
- [Sanket Dasgupta](mailto:sanketdasgupta@gmail.com)
- Sergi
- [Scott Stevenson](mailto:scott@stevenson.io)
- Shantanu
- [shaoran](mailto:shaoran@sakuranohana.org)
- [Shinya Fujino](mailto:shf0811@gmail.com)
- springstan
- [Stavros Korokithakis](mailto:hi@stavros.io)

- [Stephen Rosen](mailto:sirosen@globus.org)

- [Steven M. Vascellaro](mailto:S.Vascellaro@gmail.com)

- [Sunil Kapil](mailto:snlkapil@gmail.com)

- [Sbastien Eustace](mailto:sebastien.eustace@gmail.com)

- [Tal Amuyal](mailto:TalAmuyal@gmail.com)

- [Terrance](mailto:git@terrance.allofti.me)

- [Thom Lu](mailto:thomas.c.lu@gmail.com)

- [Thomas Grainger](mailto:tagrain@gmail.com)

- [Tim Gates](mailto:tim.gates@iress.com)

- [Tim Swast](mailto:swast@google.com)

- [Timo](mailto:timo_tk@hotmail.com)

- Toby Fleming

- [Tom Christie](mailto:tom@tomchristie.com)

- [Tony Narlock](mailto:tony@git-pull.com)

- [Tsuyoshi Hombashi](mailto:tsuyoshi.hombashi@gmail.com)

- [Tushar Chandra](mailto:tusharchandra2018@u.northwestern.edu)

- [Tzu-ping Chung](mailto:uranusjr@gmail.com)

- [Utsav Shah](mailto:ukshah2@illinois.edu)

- utsav-dbx

- vezeli

- [Ville Skytt](mailto:ville.skytta@iki.fi)

- [Vishwas B Sharma](mailto:sharma.vishwas88@gmail.com)

- [Vlad Emelianov](mailto:volshebnyi@gmail.com)

- [williamfzc](mailto:178894043@qq.com)

- [wouter bolsterlee](mailto:wouter@bolsterl.ee)

- Yazdan

- [Yngve Hiseth](mailto:yngve@hoiseth.net)

- [Yurii Karabas](mailto:1998uriyyo@gmail.com)

- [Zac Hatfield-Dodds](mailto:zac@zhd.dev)

```{include} ../AUTHORS.md

```

```{include} ../CHANGES.md

```

[flake8]
max-line-length = 88
extend-ignore = E203
--- [Deterministic header]
+++ [Deterministic header]
@@ -1,6 +1,5 @@
-def abc ():
-    return ["hello", "world",
-          "!"]
+def abc():
+   return ["hello", "world", "!"]

-print("Incorrect formatting"

```
-)
+
+print("Incorrect formatting")
{
"cells": [
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
   "tags": []
  },
  "outputs": [],
  "source": [
   "%%time\n",
   "\n",
   "print('foo')"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {},
  "outputs": [],
  "source": []
 }
],
"metadata": {
 "interpreter": {
  "hash": "e758f3098b5b55f4d87fe30bbdc1367f20f246b483f96267ee70e6c40cb185d8"
 },
 "kernelspec": {
  "display_name": "Python 3.8.10 64-bit ('black': venv)",
  "name": "python3"
 },
 "language_info": {
  "name": "python",
  "version": ""
 }
},
"nbformat": 4,
"nbformat_minor": 4
}
```

File collection and discovery

You can directly pass _Black_ files, but you can also pass directories and _Black_ will walk them, collecting files to format. It determines what files to format or skip automatically using the inclusion and exclusion regexes and as well their modification time.

Ignoring unmodified files

Black remembers files it has already formatted, unless the `--diff` flag is used or
code is passed via standard input. This information is stored per-user. The exact
location of the file depends on the _Black_ version and the system on which _Black_ is
run. The file is non-portable. The standard location on common operating systems is:

- Windows:
  `C:\\Users\<username>\AppData\Local\black\black\Cache\<version>\cache.<line-length>.<file-mode>.pickle`
- macOS:
  `/Users/<username>/Library/Caches/black/<version>/cache.<line-length>.<file-mode>.pickle`
- Linux:
  `/home/<username>/.cache/black/<version>/cache.<line-length>.<file-mode>.pickle`

`file-mode` is an int flag that determines whether the file was formatted as 3.6+ only,
as .pyi, and whether string normalization was omitted.

To override the location of these files on all systems, set the environment variable
`BLACK_CACHE_DIR` to the preferred location. Alternatively on macOS and Linux, set
`XDG_CACHE_HOME` to your preferred location. For example, if you want to put the cache
in the directory you're running _Black_ from, set `BLACK_CACHE_DIR=.cache/black`.
Black will then write the above files to `.cache/black`. Note that `BLACK_CACHE_DIR`
will take precedence over `XDG_CACHE_HOME` if both are set.

.gitignore

If `--exclude` is not set, _Black_ will automatically ignore files and directories in
`.gitignore` file(s), if present.

If you want _Black_ to continue using `.gitignore` while also configuring the exclusion
rules, please use `--extend-exclude`.
Usage and Configuration

```{toctree}
---
hidden:
---

the_basics
file_collection_and_discovery
black_as_a_server
black_docker_image
```

Sometimes, running _Black_ with its defaults and passing filepaths to it just won't cut
it. Passing each file using paths will become burdensome, and maybe you would like
Black to not touch your files and just output diffs. And yes, you _can_ tweak certain

parts of _Black_'s style, but please know that configurability in this area is
purposefully limited.

Using many of these more advanced features of _Black_ will require some configuration.
Configuration that will either live on the command line or in a TOML configuration file.

This section covers features of _Black_ and configuring _Black_ in detail:

- {doc}`The basics <./the_basics>`
- {doc}`File collection and discovery <file_collection_and_discovery>`
- {doc}`Black as a server (blackd) <./black_as_a_server>`
- {doc}`Black Docker image <./black_docker_image>`
name: changelog

on:
 pull_request:
   types: [opened, synchronize, labeled, unlabeled, reopened]

permissions:
 contents: read

jobs:
 build:
   name: Changelog Entry Check

   runs-on: ubuntu-latest

   steps:
     - uses: actions/checkout@v3

     - name: Grep CHANGES.md for PR number
       if: contains(github.event.pull_request.labels.*.name, 'skip news') != true
       run: |
         grep -Pz "\((\n\s*)?#\${{ github.event.pull_request.number }}(\n\s*)?\)" CHANGES.md || \
         (echo "Please add '(#\${{ github.event.pull_request.number }})' change line to CHANGES.md (or if
appropriate, ask a maintainer to add the 'skip news' label)" && \
           exit 1)
---
name: Documentation
about: Report a problem with or suggest something for the documentation
title: ""
labels: "T: documentation"
assignees: ""

Is this related to a problem? Please describe.

<!-- A clear and concise description of what the problem is.

e.g. I'm always frustrated when [...] / I wished that [...] -->

Describe the solution you'd like

<!-- A clear and concise description of what you want to
happen or see changed. -->

Describe alternatives you've considered

<!-- A clear and concise description of any
alternative solutions or features you've considered. -->

Additional context

<!-- Add any other context or screenshots about the issue
here. -->

```
{
"cells": [
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
   "tags": []
  },
  "outputs": [],
  "source": [
   "%%time\n",
   "\n",
   "print(\"foo\")"
  ]
 },
 {
  "cell_type": "markdown",
  "metadata": {},
  "source": [
   "This notebook should not be reformatted"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {},
  "outputs": [],
  "source": []
 }
],
"metadata": {
 "interpreter": {
```

    "hash": "e758f3098b5b55f4d87fe30bbdc1367f20f246b483f96267ee70e6c40cb185d8"
   },
   "kernelspec": {
    "display_name": "Python 3.8.10 64-bit ('black': venv)",
    "name": "python3"
   },
   "language_info": {
    "name": "python",
    "version": ""
   }
  },
  "nbformat": 4,
  "nbformat_minor": 4
  }
[tool.pylint.format]
max-line-length = "88"
Editor integration

Emacs

Options include the following:

- [wbolster/emacs-python-black](https://github.com/wbolster/emacs-python-black)
- [proofit404/blacken](https://github.com/pythonic-emacs/blacken)
- [Elpy](https://github.com/jorgenschaefer/elpy).

PyCharm/IntelliJ IDEA

There are three different ways you can use _Black_ from PyCharm:

1. As local server using the BlackConnect plugin
1. As external tool
1. As file watcher

The first option is the simplest to set up and formats the fastest (by spinning up
{doc}`Black's HTTP server </usage_and_configuration/black_as_a_server>`, avoiding the
startup cost on subsequent formats), but if you would prefer to not install a
third-party plugin or blackd's extra dependencies, the other two are also great options.

As local server

1. Install _Black_ with the `d` extra.

   ```console
   $ pip install 'black[d]'
   ```

1. Install

[BlackConnect IntelliJ IDEs plugin](https://plugins.jetbrains.com/plugin/14321-blackconnect).

1. Open plugin configuration in PyCharm/IntelliJ IDEA

   On macOS:

   `PyCharm -> Preferences -> Tools -> BlackConnect`

   On Windows / Linux / BSD:

   `File -> Settings -> Tools -> BlackConnect`

1. In `Local Instance (shared between projects)` section:

   1. Check `Start local blackd instance when plugin loads`.
   1. Press the `Detect` button near `Path` input. The plugin should detect the `blackd`
      executable.

1. In `Trigger Settings` section check `Trigger on code reformat` to enable code
   reformatting with _Black_.

1. Format the currently opened file by selecting `Code -> Reformat Code` or using a
   shortcut.

1. Optionally, to run _Black_ on every file save:

   - In `Trigger Settings` section of plugin configuration check
     `Trigger when saving changed files`.

As external tool

1. Install `black`.

   ```console
   $ pip install black
   ```

1. Locate your `black` installation folder.

   On macOS / Linux / BSD:

   ```console
   $ which black
   /usr/local/bin/black  # possible location
   ```

   On Windows:

```console
$ where black
%LocalAppData%\Programs\Python\Python36-32\Scripts\black.exe  # possible location
```

Note that if you are using a virtual environment detected by PyCharm, this is an
unneeded step. In this case the path to `black` is `$PyInterpreterDirectory$/black`.

1. Open External tools in PyCharm/IntelliJ IDEA

   On macOS:

   `PyCharm -> Preferences -> Tools -> External Tools`

   On Windows / Linux / BSD:

   `File -> Settings -> Tools -> External Tools`

1. Click the + icon to add a new external tool with the following values:

   - Name: Black
   - Description: Black is the uncompromising Python code formatter.
   - Program: \<install_location_from_step_2>
   - Arguments: `"$FilePath$"`

1. Format the currently opened file by selecting `Tools -> External Tools -> black`.

   - Alternatively, you can set a keyboard shortcut by navigating to
     `Preferences or Settings -> Keymap -> External Tools -> External Tools - Black`.

As file watcher

1. Install `black`.

   ```console
   $ pip install black
   ```

1. Locate your `black` installation folder.

   On macOS / Linux / BSD:

   ```console
   $ which black
   /usr/local/bin/black  # possible location
   ```

   On Windows:

```console
$ where black
%LocalAppData%\Programs\Python\Python36-32\Scripts\black.exe  # possible location
```

Note that if you are using a virtual environment detected by PyCharm, this is an
unneeded step. In this case the path to `black` is `$PyInterpreterDirectory$/black`.

1. Make sure you have the
   [File Watchers](https://plugins.jetbrains.com/plugin/7177-file-watchers) plugin
   installed.
1. Go to `Preferences or Settings -> Tools -> File Watchers` and click `+` to add a new
   watcher:
   - Name: Black
   - File type: Python
   - Scope: Project Files
   - Program: \<install_location_from_step_2>
   - Arguments: `$FilePath$`
   - Output paths to refresh: `$FilePath$`
   - Working directory: `$ProjectFileDir$`

   - In Advanced Options
    - Uncheck "Auto-save edited files to trigger the watcher"
    - Uncheck "Trigger the watcher on external changes"

Wing IDE

Wing IDE supports `black` via **Preference Settings** for system wide settings and
Project Properties for per-project or workspace specific settings, as explained in
the Wing documentation on
[Auto-Reformatting](https://wingware.com/doc/edit/auto-reformatting). The detailed
procedure is:

Prerequistes

- Wing IDE version 8.0+

- Install `black`.

```console
$ pip install black
```

- Make sure it runs from the command line, e.g.

```console
$ black --help
```

```
```

Preference Settings

If you want Wing IDE to always reformat with `black` for every project, follow these
steps:

1. In menubar navigate to `Edit -> Preferences -> Editor -> Reformatting`.

1. Set **Auto-Reformat** from `disable` (default) to `Line after edit` or
 `Whole files before save`.

1. Set **Reformatter** from `PEP8` (default) to `Black`.

Project Properties

If you want to just reformat for a specific project and not intervene with Wing IDE
global setting, follow these steps:

1. In menubar navigate to `Project -> Project Properties -> Options`.

1. Set **Auto-Reformat** from `Use Preferences setting` (default) to `Line after edit`
 or `Whole files before save`.

1. Set **Reformatter** from `Use Preferences setting` (default) to `Black`.

Vim

Official plugin

Commands and shortcuts:

- `:Black` to format the entire file (ranges not supported);
 - you can optionally pass `target_version=<version>` with the same values as in the
   command line.
- `:BlackUpgrade` to upgrade _Black_ inside the virtualenv;
- `:BlackVersion` to get the current version of _Black_ in use.

Configuration:

- `g:black_fast` (defaults to `0`)
- `g:black_linelength` (defaults to `88`)
- `g:black_skip_string_normalization` (defaults to `0`)
- `g:black_skip_magic_trailing_comma` (defaults to `0`)
- `g:black_virtualenv` (defaults to `~/.vim/black` or `~/.local/share/nvim/black`)
- `g:black_use_virtualenv` (defaults to `1`)
- `g:black_target_version` (defaults to `""`)
- `g:black_quiet` (defaults to `0`)

- `g:black_preview` (defaults to `0`)

Installation

This plugin **requires Vim 7.0+ built with Python 3.7+ support**. It needs Python 3.7 to be able to run _Black_ inside the Vim process which is much faster than calling an external command.

`vim-plug`

To install with [vim-plug](https://github.com/junegunn/vim-plug):

Black's `stable` branch tracks official version updates, and can be used to simply follow the most recent stable version.

```
Plug 'psf/black', { 'branch': 'stable' }
```

Another option which is a bit more explicit and offers more control is to use `vim-plug`'s `tag` option with a shell wildcard. This will resolve to the latest tag which matches the given pattern.

The following matches all stable versions (see the [Release Process](../contributing/release_process.md) section for documentation of version scheme used by Black):

```
Plug 'psf/black', { 'tag': '*.*.*' }
```

and the following demonstrates pinning to a specific year's stable style (2022 in this case):

```
Plug 'psf/black', { 'tag': '22.*.*' }
```

Vundle

or with [Vundle](https://github.com/VundleVim/Vundle.vim):

```
Plugin 'psf/black'
```

and execute the following in a terminal:

```console
$ cd ~/.vim/bundle/black
$ git checkout origin/stable -b stable
```

Arch Linux

On Arch Linux, the plugin is shipped with the
[`python-black`](https://archlinux.org/packages/community/any/python-black/) package, so
you can start using it in Vim after install with no additional setup.

Vim 8 Native Plugin Management

or you can copy the plugin files from
[plugin/black.vim](https://github.com/psf/black/blob/stable/plugin/black.vim) and
[autoload/black.vim](https://github.com/psf/black/blob/stable/autoload/black.vim).

```
mkdir -p ~/.vim/pack/python/start/black/plugin
mkdir -p ~/.vim/pack/python/start/black/autoload
curl https://raw.githubusercontent.com/psf/black/stable/plugin/black.vim -o
~/.vim/pack/python/start/black/plugin/black.vim
curl https://raw.githubusercontent.com/psf/black/stable/autoload/black.vim -o
~/.vim/pack/python/start/black/autoload/black.vim
```

Let me know if this requires any changes to work with Vim 8's builtin `packadd`, or
Pathogen, and so on.

Usage

On first run, the plugin creates its own virtualenv using the right Python version and
automatically installs _Black_. You can upgrade it later by calling `:BlackUpgrade` and
restarting Vim.

If you need to do anything special to make your virtualenv work and install _Black_ (for
example you want to run a version from main), create a virtualenv manually and point
`g:black_virtualenv` to it. The plugin will use it.

If you would prefer to use the system installation of _Black_ rather than a virtualenv,
then add this to your vimrc:

```
let g:black_use_virtualenv = 0
```

Note that the `:BlackUpgrade` command is only usable and useful with a virtualenv, so
when the virtualenv is not in use, `:BlackUpgrade` is disabled. If you need to upgrade

the system installation of _Black_, then use your system package manager or pip--
whatever tool you used to install _Black_ originally.

To run _Black_ on save, add the following lines to `.vimrc` or `init.vim`:

```
augroup black_on_save
 autocmd!
 autocmd BufWritePre *.py Black
augroup end
```

To run _Black_ on a key press (e.g. F9 below), add this:

```
nnoremap <F9> :Black<CR>
```

Troubleshooting

How to get Vim with Python 3.6? On Ubuntu 17.10 Vim comes with Python 3.6 by
default. On macOS with Homebrew run: `brew install vim`. When building Vim from source,
use: `./configure --enable-python3interp=yes`. There's many guides online how to do
this.

I get an import error when using _Black_ from a virtual environment: If you get an
error message like this:

```text
Traceback (most recent call last):
 File "<string>", line 63, in <module>
 File "/home/gui/.vim/black/lib/python3.7/site-packages/black.py", line 45, in <module>
   from typed_ast import ast3, ast27
 File "/home/gui/.vim/black/lib/python3.7/site-packages/typed_ast/ast3.py", line 40, in <module>
   from typed_ast import _ast3
ImportError: /home/gui/.vim/black/lib/python3.7/site-packages/typed_ast/_ast3.cpython-37m-x86_64-linux-gnu.so:
undefined symbool: PyExc_KeyboardInterrupt
```

Then you need to install `typed_ast` directly from the source code. The error happens
because `pip` will download [Python wheels](https://pythonwheels.com/) if they are
available. Python wheels are a new standard of distributing Python packages and packages
that have Cython and extensions written in C are already compiled, so the installation
is much more faster. The problem here is that somehow the Python environment inside Vim
does not match with those already compiled C extensions and these kind of errors are the
result. Luckily there is an easy fix: installing the packages from the source code.

The package that causes problems is:

- [typed-ast](https://pypi.org/project/typed-ast/)

Now remove those two packages:

```console
$ pip uninstall typed-ast -y
```

And now you can install them with:

```console
$ pip install --no-binary :all: typed-ast
```

The C extensions will be compiled and now Vim's Python environment will match. Note that you need to have the GCC compiler and the Python development files installed (on Ubuntu/Debian do `sudo apt-get install build-essential python3-dev`).

If you later want to update _Black_, you should do it like this:

```console
$ pip install -U black --no-binary typed-ast
```

With ALE

1. Install [`ale`](https://github.com/dense-analysis/ale)

1. Install `black`

1. Add this to your vimrc:

   ```vim
   let g:ale_fixers = {}
   let g:ale_fixers.python = ['black']
   ```

Gedit

gedit is the default text editor of the GNOME, Unix like Operating Systems. Open gedit as

```console
$ gedit <file_name>
```

1. `Go to edit > preferences > plugins`

1. Search for `external tools` and activate it.
1. In `Tools menu -> Manage external tools`
1. Add a new tool using `+` button.
1. Copy the below content to the code window.

```console
#!/bin/bash
Name=$GEDIT_CURRENT_DOCUMENT_NAME
black $Name
```

- Set a keyboard shortcut if you like, Ex. `ctrl-B`
- Save: `Nothing`
- Input: `Nothing`
- Output: `Display in bottom pane` if you like.
- Change the name of the tool if you like.

Use your keyboard shortcut or `Tools -> External Tools` to use your new tool. When you
close and reopen your File, _Black_ will be done with its job.

Visual Studio Code

- Use the
 [Python extension](https://marketplace.visualstudio.com/items?itemName=ms-python.python)
 ([instructions](https://code.visualstudio.com/docs/python/editing#_formatting)).

- Alternatively the pre-release
 [Black Formatter](https://marketplace.visualstudio.com/items?itemName=ms-python.black-formatter)
 extension can be used which runs a [Language Server Protocol](https://langserver.org/)
 server for Black. Formatting is much more responsive using this extension, **but the
 minimum supported version of Black is 22.3.0**.

SublimeText 3

Use [sublack plugin](https://github.com/jgirardet/sublack).

Python LSP Server

If your editor supports the [Language Server Protocol](https://langserver.org/) (Atom,
Sublime Text, Visual Studio Code and many more), you can use the
[Python LSP Server](https://github.com/python-lsp/python-lsp-server) with the
[python-lsp-black](https://github.com/python-lsp/python-lsp-black) plugin.

Atom/Nuclide

Use [python-black](https://atom.io/packages/python-black) or
[formatters-python](https://atom.io/packages/formatters-python).

Gradle (the build tool)

Use the [Spotless](https://github.com/diffplug/spotless/tree/main/plugin-gradle) plugin.

Kakoune

Add the following hook to your kakrc, then run _Black_ with `:format`.

```
hook global WinSetOption filetype=python %{
    set-option window formatcmd 'black -q  -'
}
```

Thonny

Use [Thonny-black-code-format](https://github.com/Franccisco/thonny-black-code-format).
name: diff-shades

on:
 push:
   branches: [main]
   paths: ["src/**", "pyproject.toml", ".github/workflows/*"]

 pull_request:
   paths: ["src/**", "pyproject.toml", ".github/workflows/*"]

concurrency:
 group: ${{ github.workflow }}-${{ github.event.pull_request.number || github.run_id }}
 cancel-in-progress: true

jobs:
 configure:
   runs-on: ubuntu-latest
   outputs:
     matrix: ${{ steps.set-config.outputs.matrix }}

   steps:
     - uses: actions/checkout@v3
     - uses: actions/setup-python@v4
       with:
         python-version: "*"

     - name: Install diff-shades and support dependencies
       run: |
         python -m pip install click packaging urllib3
         python -m pip install https://github.com/ichard26/diff-shades/archive/stable.zip

```yaml
    - name: Calculate run configuration & metadata
      id: set-config
      env:
        GITHUB_TOKEN: ${{ github.token }}
      run: >
        python scripts/diff_shades_gha_helper.py config ${{ github.event_name }} ${{ matrix.mode }}


  analysis:
    name: analysis / ${{ matrix.mode }}
    needs: configure
    runs-on: ubuntu-latest
    env:
      HATCH_BUILD_HOOKS_ENABLE: "1"
      # Clang is less picky with the C code it's given than gcc (and may
      # generate faster binaries too).
      CC: clang-12
    strategy:
      fail-fast: false
      matrix:
        include: ${{ fromJson(needs.configure.outputs.matrix )}}


    steps:
      - name: Checkout this repository (full clone)
        uses: actions/checkout@v3
        with:
          # The baseline revision could be rather old so a full clone is ideal.
          fetch-depth: 0


      - uses: actions/setup-python@v4
        with:
          python-version: "*"


      - name: Install diff-shades and support dependencies
        run: |
          python -m pip install https://github.com/ichard26/diff-shades/archive/stable.zip
          python -m pip install click packaging urllib3
          # After checking out old revisions, this might not exist so we'll use a copy.
          cat scripts/diff_shades_gha_helper.py > helper.py
          git config user.name "diff-shades-gha"
          git config user.email "diff-shades-gha@example.com"


      - name: Attempt to use cached baseline analysis
        id: baseline-cache
        uses: actions/cache@v3
        with:
          path: ${{ matrix.baseline-analysis }}
          key: ${{ matrix.baseline-cache-key }}
```

```yaml
- name: Build and install baseline revision
  if: steps.baseline-cache.outputs.cache-hit != 'true'
  env:
    GITHUB_TOKEN: ${{ github.token }}
  run: >
    ${{ matrix.baseline-setup-cmd }}
    && python -m pip install .

- name: Analyze baseline revision
  if: steps.baseline-cache.outputs.cache-hit != 'true'
  run: >
    diff-shades analyze -v --work-dir projects-cache/
    ${{ matrix.baseline-analysis }} ${{ matrix.force-flag }}

- name: Build and install target revision
  env:
    GITHUB_TOKEN: ${{ github.token }}
  run: >
    ${{ matrix.target-setup-cmd }}
    && python -m pip install .

- name: Analyze target revision
  run: >
    diff-shades analyze -v --work-dir projects-cache/
    ${{ matrix.target-analysis }} --repeat-projects-from
    ${{ matrix.baseline-analysis }} ${{ matrix.force-flag }}

- name: Generate HTML diff report
  run: >
    diff-shades --dump-html diff.html compare --diff
    ${{ matrix.baseline-analysis }} ${{ matrix.target-analysis }}

- name: Upload diff report
  uses: actions/upload-artifact@v3
  with:
    name: ${{ matrix.mode }}-diff.html
    path: diff.html

- name: Upload baseline analysis
  uses: actions/upload-artifact@v3
  with:
    name: ${{ matrix.baseline-analysis }}
    path: ${{ matrix.baseline-analysis }}

- name: Upload target analysis
  uses: actions/upload-artifact@v3
  with:
    name: ${{ matrix.target-analysis }}
```

```
        path: ${{ matrix.target-analysis }}

    - name: Generate summary file (PR only)
      if: github.event_name == 'pull_request' && matrix.mode == 'preview-changes'
      run: >
        python helper.py comment-body
        ${{ matrix.baseline-analysis }} ${{ matrix.target-analysis }}
        ${{ matrix.baseline-sha }} ${{ matrix.target-sha }}
        ${{ github.event.pull_request.number }}

    - name: Upload summary file (PR only)
      if: github.event_name == 'pull_request' && matrix.mode == 'preview-changes'
      uses: actions/upload-artifact@v3
      with:
        name: .pr-comment.json
        path: .pr-comment.json

    - name: Verify zero changes (PR only)
      if: matrix.mode == 'assert-no-changes'
      run: >
        diff-shades compare --check ${{ matrix.baseline-analysis }} ${{ matrix.target-analysis }}
        || (echo "Please verify you didn't change the stable code style unintentionally!" && exit 1)

    - name: Check for failed files for target revision
      # Even if the previous step failed, we should still check for failed files.
      if: always()
      run: >
        diff-shades show-failed --check --show-log ${{ matrix.target-analysis }}
```

Black Docker image

Official _Black_ Docker images are available on
[Docker Hub](https://hub.docker.com/r/pyfound/black).

Black images with the following tags are available:

- release numbers, e.g. `21.5b2`, `21.6b0`, `21.7b0` etc.\
  Recommended for users who want to use a particular version of _Black_.
- `latest_release` - tag created when a new version of _Black_ is released.\
  Recommended for users who want to use released versions of _Black_. It maps to [the latest
  release](https://github.com/psf/black/releases/latest)
  of _Black_.
- `latest_prerelease` - tag created when a new alpha (prerelease) version of _Black_ is
  released.\
  Recommended for users who want to preview or test alpha versions of _Black_. Note that
  the most recent release may be newer than any prerelease, because no prereleases are created
  before most releases.
- `latest` - tag used for the newest image of _Black_.\
  Recommended for users who always want to use the latest version of _Black_, even before

it is released.

There is one more tag used for _Black_ Docker images - `latest_non_release`. It is
created for all unreleased
[commits on the `main` branch](https://github.com/psf/black/commits/main). This tag is
not meant to be used by external users.

Usage

A permanent container doesn't have to be created to use _Black_ as a Docker image. It's
enough to run _Black_ commands for the chosen image denoted as `:tag`. In the below
examples, the `latest_release` tag is used. If `:tag` is omitted, the `latest` tag will
be used.

More about _Black_ usage can be found in
[Usage and Configuration: The basics](./the_basics.md).

Check Black version

```console
$ docker run --rm pyfound/black:latest_release black --version
```

Check code

```console
$ docker run --rm --volume $(pwd):/src --workdir /src pyfound/black:latest_release black --check .
```

Remark: besides [regular _Black_ exit codes](./the_basics.md) returned by `--check`
option, [Docker exit codes](https://docs.docker.com/engine/reference/run/#exit-status)
should also be considered.
name: Documentation

on: [push, pull_request]

permissions:
 contents: read

jobs:
 build:
   # We want to run on external PRs, but not on our own internal PRs as they'll be run
   # by the push to the branch. Without this if check, checks are duplicated since
   # internal PRs match both the push and pull_request events.
   if:
     github.event_name == 'push' || github.event.pull_request.head.repo.full_name !=
     github.repository

```yaml
    strategy:
      fail-fast: false
      matrix:
        os: [ubuntu-latest, windows-latest]

    runs-on: ${{ matrix.os }}
    steps:
      - uses: actions/checkout@v3

      - name: Set up latest Python
        uses: actions/setup-python@v4
        with:
          python-version: "*"

      - name: Install dependencies
        run: |
          python -m pip install --upgrade pip setuptools wheel
          python -m pip install -e ".[d]"
          python -m pip install -r "docs/requirements.txt"

      - name: Build documentation
        run: sphinx-build -a -b html -W --keep-going docs/ docs/_build
```
The MIT License (MIT)

Copyright (c) 2018 ukasz Langa

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```
proseWrap: always
printWidth: 88
endOfLine: auto
[project]
name = "test"
```

```
version = "1.0.0"

[tool.black]
line-length = 79
name: Publish to PyPI

on:
 release:
  types: [published]

permissions:
 contents: read

jobs:
 main:
  name: sdist + pure wheel
  runs-on: ubuntu-latest

  steps:
   - uses: actions/checkout@v3

   - name: Set up latest Python
     uses: actions/setup-python@v4
     with:
      python-version: "*"

   - name: Install latest pip, build, twine
     run: |
       python -m pip install --upgrade --disable-pip-version-check pip
       python -m pip install --upgrade build twine

   - name: Build wheel and source distributions
     run: python -m build

   - name: Upload to PyPI via Twine
     env:
      TWINE_PASSWORD: ${{ secrets.PYPI_TOKEN }}
     run: twine upload --verbose -u '__token__' dist/*

 mypyc:
  name: mypyc wheels (${{ matrix.name }})
  runs-on: ${{ matrix.os }}
  strategy:
   fail-fast: false
   matrix:
    include:
     - os: ubuntu-latest
       name: linux-x86_64
```

```yaml
        - os: windows-2019
          name: windows-amd64
        - os: macos-11
          name: macos-x86_64
          macos_arch: "x86_64"
        - os: macos-11
          name: macos-arm64
          macos_arch: "arm64"
        - os: macos-11
          name: macos-universal2
          macos_arch: "universal2"

    steps:
      - uses: actions/checkout@v3

      - name: Build wheels via cibuildwheel
        uses: pypa/cibuildwheel@v2.11.4
        env:
          CIBW_ARCHS_MACOS: "${{ matrix.macos_arch }}"

      - name: Upload wheels as workflow artifacts
        uses: actions/upload-artifact@v3
        with:
          name: ${{ matrix.name }}-mypyc-wheels
          path: ./wheelhouse/*.whl

      - name: Upload wheels to PyPI via Twine
        env:
          TWINE_PASSWORD: ${{ secrets.PYPI_TOKEN }}
        run: pipx run twine upload --verbose -u '__token__' wheelhouse/*.whl

  update-stable-branch:
    name: Update stable branch
    needs: [main, mypyc]
    runs-on: ubuntu-latest
    permissions:
      contents: write

    steps:
      - name: Checkout stable branch
        uses: actions/checkout@v3
        with:
          ref: stable
          fetch-depth: 0

      - name: Update stable branch to release tag & push
        run: |
          git reset --hard ${{ github.event.release.tag_name }}
```

```
      git push
[format]
max-line-length = 88
file_input
 decorated
   decorator
     AT
'@'
     NAME
'dataclass'
     NEWLINE
'\n'
   /decorator
   classdef
     NAME
'class'
     NAME
' '
'DebugVisitor'
     LPAR
'('
     power
       NAME
'Visitor'
       trailer
         LSQB
'['
         NAME
'T'
         RSQB
']'
       /trailer
     /power
     RPAR
')'
     COLON
':'
     suite
     NEWLINE
'\n'
       INDENT
''
       simple_stmt
         expr_stmt
           NAME
'   '
'tree_depth'
           annassign
```

COLON

':'

          NAME

' '

'int'

          EQUAL

' '

'='

          NUMBER

' '

'0'

        /annassign

      /expr_stmt

      NEWLINE

'\n'

    /simple_stmt

    funcdef

      NAME

'\n   '

'def'

      NAME

' '

'visit_default'

      parameters

        LPAR

'('

        typedargslist

          NAME

'self'

          COMMA

','

          tname

            NAME

' '

'node'

            COLON

':'

            NAME

' '

'LN'

          /tname

        /typedargslist

        RPAR

')'

      /parameters

      RARROW

' '

'->'

power
NAME
' '

'Iterator'
trailer
LSQB
'['
NAME
'T'
RSQB
']'
/trailer
/power
COLON
':'
suite
NEWLINE
'\n'
INDENT
''
simple_stmt
expr_stmt
NAME
'   '
'indent'
EQUAL
' '
'='
term
STRING
' '
"' '"
STAR
' '
'*'
atom
LPAR
' '
'('
term
NUMBER
'2'
STAR
' '
'*'
power
NAME
' '

'self'
                    trailer
                          DOT
'.'
                        NAME
'tree_depth'
                        /trailer
                    /power
                /term
            RPAR
')'
            /atom
        /term
      /expr_stmt
      NEWLINE
'\n'
      /simple_stmt
      if_stmt
      NAME
'   '
'if'
        power
        NAME
' '
'isinstance'
          trailer
          LPAR
'('
          arglist
            NAME
'node'
          COMMA
','
            NAME
' '
'Node'
          /arglist
          RPAR
')'
        /trailer
      /power
      COLON
':'
      suite
        NEWLINE
'\n'
        INDENT
"

simple_stmt
             expr_stmt
              NAME
'   '
'_type'
             EQUAL
' '
'='
             power
              NAME
' '
'type_repr'
              trailer
               LPAR
'('
               power
                NAME
'node'
                trailer
                 DOT
'.'
                 NAME
'type'
                /trailer
               /power
               RPAR
')'
              /trailer
             /power
            /expr_stmt
            NEWLINE
'\n'
           /simple_stmt
           simple_stmt
            power
             NAME
'   '
'out'
             trailer
              LPAR
'('
              arglist
               STRING
"f'{indent}{_type}'"
               COMMA
','
               argument
                NAME

```
                        ' '
                        'fg'
                                        EQUAL
                        '='
                                        STRING
                        "'yellow'"
                                         /argument
                                    /arglist
                                RPAR
                        ')'
                                 /trailer
                            /power
                                NEWLINE
                        '\n'
                            /simple_stmt
                        simple_stmt
                        expr_stmt
                         power
                            NAME
                        '          '
                        'self'
                                trailer
                                DOT
                        '.'
                                NAME
                        'tree_depth'
                                /trailer
                            /power
                            PLUSEQUAL
                        ' '
                        '+='
                            NUMBER
                        ' '
                        '1'
                            /expr_stmt
                            NEWLINE
                        '\n'
                            /simple_stmt
                        for_stmt
                            NAME
                        '          '
                        'for'
                            NAME
                        ' '
                        'child'
                            NAME
                        ' '
                        'in'
```

```
            power
              NAME
' '
'node'
              trailer
                DOT
'.'
                NAME
'children'
              /trailer
            /power
            COLON
':'
            suite
              NEWLINE
'\n'
              INDENT
''
              simple_stmt
                yield_expr
                  NAME
'            '
'yield'
                  yield_arg
                    NAME
' '
'from'
                    power
                      NAME
' '
'self'
                      trailer
                        DOT
'.'
                        NAME
'visit'
                      /trailer
                      trailer
                        LPAR
'('
                        NAME
'child'
                        RPAR
')'
                      /trailer
                    /power
                  /yield_arg
                /yield_expr
```

NEWLINE
'\n'
                /simple_stmt
                DEDENT
''
                /suite
            /for_stmt
            simple_stmt
            expr_stmt
            power
                NAME
'\n   '
'self'
                trailer
                    DOT
'.'
                    NAME
'tree_depth'
                /trailer
            /power
            MINEQUAL
' '
'-='
                NUMBER
' '
'1'
            /expr_stmt
            NEWLINE
'\n'
            /simple_stmt
            simple_stmt
            power
                NAME
'   '
'out'
                trailer
                LPAR
'('
                arglist
                STRING
"f'{indent}/{_type}'"
                COMMA
','
                argument
                NAME
' '
'fg'
                EQUAL

'='

                    STRING
'"yellow"'

                    /argument
                    COMMA
','

                    argument
                     NAME
' '

'bold'

                     EQUAL
'='

                     NAME
'False'

                     /argument
                    /arglist
                     RPAR
')'

                  /trailer
               /power
              NEWLINE
'\n'

            /simple_stmt
           DEDENT
''

          /suite
          NAME
'   '

'else'

          COLON
':'

          suite
           NEWLINE
'\n'

            INDENT
''

            simple_stmt
            expr_stmt
             NAME
'   '

'_type'

            EQUAL
''

'='

               power
                NAME
''

'token'

trailer
                      DOT
'.'
                      NAME
'tok_name'
                    /trailer
                    trailer
                      DOT
'.'
                      NAME
'get'
                    /trailer
                    trailer
                      LPAR
'('
                      arglist
                        power
                          NAME
'node'
                          trailer
                            DOT
'.'
                            NAME
'type'
                          /trailer
                        /power
                        COMMA
','
                        power
                          NAME
' '
'str'
                          trailer
                            LPAR
'('
                            power
                              NAME
'node'
                              trailer
                                DOT
'.'
                                NAME
'type'
                              /trailer
                            /power
                            RPAR
')'
                          /trailer

/power
                        /arglist
                        RPAR
')'
                        /trailer
                      /power
                    /expr_stmt
                    NEWLINE
'\n'
              /simple_stmt
            simple_stmt
            power
              NAME
'   '
'out'
              trailer
                LPAR
'('
                arglist
                  STRING
"f'{indent}{_type}'"
                  COMMA
','
                  argument
                    NAME
' '
'fg'
                    EQUAL
'='
                    STRING
"'blue'"
                  /argument
                  COMMA
','
                  argument
                    NAME
' '
'nl'
                    EQUAL
'='
                    NAME
'False'
                  /argument
                /arglist
                RPAR
')'
              /trailer
            /power

```
        NEWLINE
'\n'
        /simple_stmt
      if_stmt
        NAME
'       '
'if'
        power
         NAME
' '
'node'
         trailer
          DOT
'.'
          NAME
'prefix'
         /trailer
        /power
        COLON
':'
        suite
         NEWLINE
'\n'
         INDENT
''
         simple_stmt
          power
           NAME
''          # We don't have to handle prefixes for `Node` objects since\n            # that delegates to the first child anyway.\n          "
'out'
           trailer
            LPAR
'('
            arglist
             STRING
"f' {node.prefix!r}'"
             COMMA
','
             argument
              NAME
' '
'fg'
              EQUAL
'='
              STRING
"'green'"
             /argument
```

COMMA
','
                          argument
                            NAME
' '
'bold'
                            EQUAL
'='
                            NAME
'False'
                          /argument
                          COMMA
','
                          argument
                            NAME
' '
'nl'
                            EQUAL
'='
                            NAME
'False'
                          /argument
                        /arglist
                        RPAR
')'
                      /trailer
                    /power
                  NEWLINE
'\n'
                /simple_stmt
              DEDENT
''
            /suite
          /if_stmt
          simple_stmt
            power
              NAME
'   '
'out'
              trailer
                LPAR
'('
                arglist
                  STRING
"f' {node.value!r}'"
                  COMMA
','
                  argument

NAME
' '
'fg'
                EQUAL
'='
                STRING
'"blue"'
                /argument
                COMMA
','
                argument
                NAME
' '
'bold'
                EQUAL
'='
                NAME
'False'
                /argument
            /arglist
            RPAR
')'
          /trailer
        /power
      NEWLINE
'\n'
        /simple_stmt
      DEDENT
''
      /suite
    /if_stmt
    DEDENT
''
    /suite
  /funcdef
  decorated
   decorator
    AT
'\n   '
'@'
    NAME
'classmethod'
    NEWLINE
'\n'
   /decorator
   funcdef
    NAME
' '
' '

'def'
        NAME
' '
'show'
        parameters
         LPAR
'('
        typedargslist
         NAME
'cls'
         COMMA
','
         tname
          NAME
' '
'code'
          COLON
':'
          NAME
' '
'str'
         /tname
        /typedargslist
        RPAR
')'
        /parameters
        RARROW
' '
'->'
        NAME
' '
'None'
        COLON
':'
        suite
         NEWLINE
'\n'
         INDENT
"
         simple_stmt
          STRING
'   '
"""Pretty-prints a given string of `code`.\n\n   Convenience method for debugging.\n   """
          NEWLINE
'\n'
         /simple_stmt
         simple_stmt
          expr_stmt

NAME

'   '

'v'

    annassign

     COLON

':'

     power

      NAME

' '

'DebugVisitor'

      trailer

       LSQB

'['

       NAME

'None'

       RSQB

']'

      /trailer

     /power

     EQUAL

' '

'='

     power

      NAME

' '

'DebugVisitor'

      trailer

       LPAR

'('

       RPAR

')'

      /trailer

     /power

    /annassign

   /expr_stmt

   NEWLINE

'\n'

  /simple_stmt

  simple_stmt

  power

   NAME

'   '

'list'

   trailer

    LPAR

'('

    power

     NAME

'v'
                    trailer
                        DOT
'.'
                        NAME
'visit'
                    /trailer
                    trailer
                        LPAR
'('
                    power
                        NAME
'lib2to3_parse'
                        trailer
                            LPAR
'('
                            NAME
'code'
                            RPAR
')'
                        /trailer
                    /power
                        RPAR
')'
                    /trailer
                /power
                    RPAR
')'
                /trailer
            /power
            NEWLINE
'\n'
        /simple_stmt
        DEDENT
''
      /suite
     /funcdef
    /decorated
    DEDENT
''
   /suite
  /classdef
 /decorated
 ENDMARKER
''
/file_input
---
name: Bug report

about: Create a report to help us improve Black's quality
title: ""
labels: "T: bug"
assignees: ""

<!--
Please make sure that the bug is not already fixed either in newer versions or the
current development version. To confirm this, you have three options:

1. Update Black's version if a newer release exists: `pip install -U black`
2. Use the online formatter at <https://black.vercel.app/?version=main>, which will use
   the latest main branch.
3. Or run _Black_ on your machine:
   - create a new virtualenv (make sure it's the same Python version);
   - clone this repository;
   - run `pip install -e .[d]`;
   - run `pip install -r test_requirements.txt`
   - make sure it's sane by running `python -m pytest`; and
   - run `black` like you did last time.
-->

Describe the bug

<!-- A clear and concise description of what the bug is. -->

To Reproduce

<!--
Minimal steps to reproduce the behavior with source code and Black's configuration.
-->

For example, take this code:

```python
this = "code"
```

And run it with these arguments:

```sh
$ black file.py --target-version py39
```

The resulting error is:

> cannot format file.py: INTERNAL ERROR: ...

Expected behavior

<!-- A clear and concise description of what you expected to happen. -->

Environment

<!-- Please complete the following information: -->

- Black's version: <!-- e.g. [main] -->
- OS and Python version: <!-- e.g. [Linux/Python 3.7.4rc1] -->

Additional context

<!-- Add any other context about the problem here. -->
--- [Deterministic header]
+++ [Deterministic header]
@@ -1,8 +1,8 @@
...
-'some_string'
-b'\\xa3'
+"some_string"
+b"\\xa3"
Name
None
True
False
1
@@ -21,99 +21,135 @@
Name1 or (Name2 and Name3) or Name4
Name1 or Name2 and Name3 or Name4
v1 << 2
1 >> v2
1 % finished
-1 + v2 - v3 * 4 ^ 5 ** v6 / 7 // 8
-((1 + v2) - (v3 * 4)) ^ (((5 ** v6) / 7) // 8)
+1 + v2 - v3 * 4 ^ 5**v6 / 7 // 8
+((1 + v2) - (v3 * 4)) ^ (((5**v6) / 7) // 8)
not great
~great
+value
-1
~int and not v1 ^ 123 + v2 | True
(~int) and (not ((v1 ^ (123 + v2)) | True))
-+really ** -confusing ** ~operator ** -precedence
-flags & ~ select.EPOLLIN and waiters.write_task is not None
++(really ** -(confusing ** ~(operator**-precedence)))
+flags & ~select.EPOLLIN and waiters.write_task is not None
lambda arg: None

lambda a=True: a

lambda a, b, c=True: a

-lambda a, b, c=True, *, d=(1 << v2), e='str': a

-lambda a, b, c=True, *vararg, d=(v1 << 2), e='str', **kwargs: a + b

+lambda a, b, c=True, *, d=(1 << v2), e="str": a

+lambda a, b, c=True, *vararg, d=(v1 << 2), e="str", **kwargs: a + b

manylambdas = lambda x=lambda y=lambda z=1: z: y(): x()

-foo = (lambda port_id, ignore_missing: {"port1": port1_resource, "port2": port2_resource}[port_id])

+foo = lambda port_id, ignore_missing: {

+   "port1": port1_resource,

+   "port2": port2_resource,

+}[port_id]

1 if True else 2

str or None if True else str or bytes or None

(str or None) if True else (str or bytes or None)

str or None if (1 if True else 2) else str or bytes or None

(str or None) if (1 if True else 2) else (str or bytes or None)

-((super_long_variable_name or None) if (1 if super_long_test_name else 2) else (str or bytes or None))

-{'2.7': dead, '3.7': (long_live or die_hard)}

-{'2.7': dead, '3.7': (long_live or die_hard), **{'3.6': verygood}}

+(

+   (super_long_variable_name or None)

+   if (1 if super_long_test_name else 2)

+   else (str or bytes or None)

+)

+{"2.7": dead, "3.7": (long_live or die_hard)}

+{"2.7": dead, "3.7": (long_live or die_hard), **{"3.6": verygood}}

{**a, **b, **c}

-{'2.7', '3.6', '3.7', '3.8', '3.9', ('4.0' if gilectomy else '3.10')}

-({'a': 'b'}, (True or False), (+value), 'string', b'bytes') or None

+{"2.7", "3.6", "3.7", "3.8", "3.9", ("4.0" if gilectomy else "3.10")}

+({"a": "b"}, (True or False), (+value), "string", b"bytes") or None

()

(1,)

(1, 2)

(1, 2, 3)

[]

[1, 2, 3, 4, 5, 6, 7, 8, 9, (10 or A), (11 or B), (12 or C)]

-[1, 2, 3,]

+[

+   1,

+   2,

+   3,

+]

[*a]

[*range(10)]

-[*a, 4, 5,]

-[4, *a, 5,]

```
-[this_is_a_very_long_variable_which_will_force_a_delimiter_split, element, another, *more]
+[
+    *a,
+    4,
+    5,
+]
+[
+    4,
+    *a,
+    5,
+]
+[
+    this_is_a_very_long_variable_which_will_force_a_delimiter_split,
+    element,
+    another,
+    *more,
+]
{i for i in (1, 2, 3)}
-{(i ** 2) for i in (1, 2, 3)}
-{(i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c'))}
-{((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3)}
+{(i**2) for i in (1, 2, 3)}
+{(i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c"))}
+{((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3)}
[i for i in (1, 2, 3)]
-[(i ** 2) for i in (1, 2, 3)]
-[(i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c'))]
-[((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3)]
+[(i**2) for i in (1, 2, 3)]
+[(i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c"))]
+[((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3)]
{i: 0 for i in (1, 2, 3)}
-{i: j for i, j in ((1, 'a'), (2, 'b'), (3, 'c'))}
+{i: j for i, j in ((1, "a"), (2, "b"), (3, "c"))}
{a: b * 2 for a, b in dictionary.items()}
{a: b * -2 for a, b in dictionary.items()}
-{k: v for k, v in
this_is_a_very_long_variable_which_will_cause_a_trailing_comma_which_breaks_the_comprehension}
+{
+    k: v
+    for k, v in
this_is_a_very_long_variable_which_will_cause_a_trailing_comma_which_breaks_the_comprehension
+}
Python3 > Python2 > COBOL
Life is Life
call()
call(arg)
-call(kwarg='hey')
```

```
-call(arg, kwarg='hey')
-call(arg, another, kwarg='hey', **kwargs)
-call(this_is_a_very_long_variable_which_will_force_a_delimiter_split, arg, another, kwarg='hey', **kwargs)  #
note: no trailing comma pre-3.6
+call(kwarg="hey")
+call(arg, kwarg="hey")
+call(arg, another, kwarg="hey", **kwargs)
+call(
+    this_is_a_very_long_variable_which_will_force_a_delimiter_split,
+    arg,
+    another,
+    kwarg="hey",
+    **kwargs
+)  # note: no trailing comma pre-3.6
call(*gidgets[:2])
call(a, *gidgets[:2])
call(**self.screen_kwargs)
call(b, **self.screen_kwargs)
lukasz.langa.pl
call.me(maybe)
-1 .real
-1.0 .real
+(1).real
+(1.0).real
....__class__
list[str]
dict[str, int]
tuple[str, ...]
+tuple[str, int, float, dict[str, int]]
tuple[
-    str, int, float, dict[str, int]
-]
-tuple[str, int, float, dict[str, int],]
+    str,
+    int,
+    float,
+    dict[str, int],
+]
very_long_variable_name_filters: t.List[
    t.Tuple[str, t.Union[str, t.List[t.Optional[str]]]],
]
xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(  # type: ignore
    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
)
xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(  # type: ignore
    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
)
-xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[
```

```
-    ..., List[SomeClass]
-] = classmethod(sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__))  # type: ignore
+xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(
+    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
+)  # type: ignore
slice[0]
slice[0:1]
slice[0:1:2]
slice[:]
slice[:-1]
@@ -137,118 +173,199 @@
numpy[-(c + 1) :, d]
numpy[:, l[-2]]
numpy[:, ::-1]
numpy[np.newaxis, :]
(str or None) if (sys.version_info[0] > (3,)) else (str or bytes or None)
-{'2.7': dead, '3.7': long_live or die_hard}
-{'2.7', '3.6', '3.7', '3.8', '3.9', '4.0' if gilectomy else '3.10'}
+{"2.7": dead, "3.7": long_live or die_hard}
+{"2.7", "3.6", "3.7", "3.8", "3.9", "4.0" if gilectomy else "3.10"}
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10 or A, 11 or B, 12 or C]
(SomeName)
SomeName
(Good, Bad, Ugly)
(i for i in (1, 2, 3))
-((i ** 2) for i in (1, 2, 3))
-((i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c')))
-(((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3))
+((i**2) for i in (1, 2, 3))
+((i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c")))
+(((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3))
(*starred,)
-{"id": "1","type": "type","started_at": now(),"ended_at": now() + timedelta(days=10),"priority":
1,"import_session_id": 1,**kwargs}
+{
+    "id": "1",
+    "type": "type",
+    "started_at": now(),
+    "ended_at": now() + timedelta(days=10),
+    "priority": 1,
+    "import_session_id": 1,
+    **kwargs,
+}
a = (1,)
-b = 1,
+b = (1,)
c = 1
d = (1,) + a + (2,)
```

```
e = (1,).count(1)
f = 1, *range(10)
g = 1, *"ten"
-what_is_up_with_those_new_coord_names = (coord_names + set(vars_to_create)) + set(vars_to_remove)
-what_is_up_with_those_new_coord_names = (coord_names | set(vars_to_create)) - set(vars_to_remove)
-result = session.query(models.Customer.id).filter(models.Customer.account_id == account_id,
models.Customer.email == email_address).order_by(models.Customer.id.asc()).all()
-result = session.query(models.Customer.id).filter(models.Customer.account_id == account_id,
models.Customer.email == email_address).order_by(models.Customer.id.asc(),).all()
+what_is_up_with_those_new_coord_names = (coord_names + set(vars_to_create)) + set(
+    vars_to_remove
+)
+what_is_up_with_those_new_coord_names = (coord_names | set(vars_to_create)) - set(
+    vars_to_remove
+)
+result = (
+    session.query(models.Customer.id)
+    .filter(
+        models.Customer.account_id == account_id, models.Customer.email == email_address
+    )
+    .order_by(models.Customer.id.asc())
+    .all()
+)
+result = (
+    session.query(models.Customer.id)
+    .filter(
+        models.Customer.account_id == account_id, models.Customer.email == email_address
+    )
+    .order_by(
+        models.Customer.id.asc(),
+    )
+    .all()
+)
 = set()
authors.ukasz.say_thanks()
mapping = {
    A: 0.25 * (10.0 / 12),
    B: 0.1 * (10.0 / 12),
    C: 0.1 * (10.0 / 12),
    D: 0.1 * (10.0 / 12),
}


+
def gen():
    yield from outside_of_generator
-    a = (yield)
-    b = ((yield))
-    c = (((yield)))
```

```
+    a = yield
+    b = yield
+    c = yield
+

async def f():
    await some.complicated[0].call(with_args=(True or (1 is not 1)))
-print(* [] or [1])
+
+
+print(*[] or [1])
print(**{1: 3} if False else {x: x for x in range(3)})
-print(* lambda x: x)
-assert(not Test),("Short message")
-assert this is ComplexTest and not requirements.fit_in_a_single_line(force=False), "Short message"
-assert(((parens is TooMany)))
-for x, in (1,), (2,), (3,): ...
-for y in (): ...
-for z in (i for i in (1, 2, 3)): ...
-for i in (call()): ...
-for j in (1 + (2 + 3)): ...
-while(this and that): ...
-for addr_family, addr_type, addr_proto, addr_canonname, addr_sockaddr in socket.getaddrinfo('google.com',
'http'):
+print(*lambda x: x)
+assert not Test, "Short message"
+assert this is ComplexTest and not requirements.fit_in_a_single_line(
+    force=False
+), "Short message"
+assert parens is TooMany
+for (x,) in (1,), (2,), (3,):
+    ...
+for y in ():
+    ...
+for z in (i for i in (1, 2, 3)):
+    ...
+for i in call():
+    ...
+for j in 1 + (2 + 3):
+    ...
+while this and that:
+    ...
+for (
+    addr_family,
+    addr_type,
+    addr_proto,
+    addr_canonname,
+    addr_sockaddr,
```

```
+) in socket.getaddrinfo("google.com", "http"):
    pass
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp in
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp not in
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp is
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp is not
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-if (
-   threading.current_thread() != threading.main_thread() and
-   threading.current_thread() != threading.main_thread() or
-   signal.getsignal(signal.SIGINT) != signal.default_int_handler
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa |
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa &
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa +
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa -
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa *
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa /
-   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-   return True
-if (
-   ~ aaaa.a + aaaa.b - aaaa.c * aaaa.d / aaaa.e | aaaa.f & aaaa.g % aaaa.h ^ aaaa.i << aaaa.k >> aaaa.l ** aaaa.m //
```

aaaa.n

-):

-    return True

-if (

-    ~ aaaaaaaa.a + aaaaaaaa.b - aaaaaaaa.c @ aaaaaaaa.d / aaaaaaaa.e | aaaaaaaa.f & aaaaaaaa.g % aaaaaaaa.h ^
aaaaaaaa.i << aaaaaaaa.k >> aaaaaaaa.l ** aaaaaaaa.m // aaaaaaaa.n

-):

-    return True

-if (

-    ~ aaaaaaaaaaaaaaaaa.a + aaaaaaaaaaaaaaaa.b - aaaaaaaaaaaaaaaa.c * aaaaaaaaaaaaaaaa.d @ aaaaaaaaaaaaaaaa.e |
aaaaaaaaaaaaaaaa.f & aaaaaaaaaaaaaaaa.g % aaaaaaaaaaaaaaaa.h ^ aaaaaaaaaaaaaaaa.i << aaaaaaaaaaaaaaaa.k >>
aaaaaaaaaaaaaaaa.l ** aaaaaaaaaaaaaaaa.m // aaaaaaaaaaaaaaaa.n

-):

-    return True

-aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa - aaaaaaaaaaaaaaaa * (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa) /
(aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)

+a = (

+   aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp

+   in qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz

+)

+a = (

+   aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp

+   not in qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz

+)

+a = (

+   aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp

+   is qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz

+)

+a = (

+   aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp

+   is not qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz

+)

+if (

+   threading.current_thread() != threading.main_thread()

+   and threading.current_thread() != threading.main_thread()

+   or signal.getsignal(signal.SIGINT) != signal.default_int_handler

+):

+   return True

+if (

+   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+ | aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+):

+   return True

+if (

+   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+   & aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+):

+   return True

```
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    + aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    - aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    * aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    / aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    ~aaaa.a + aaaa.b - aaaa.c * aaaa.d / aaaa.e
+    | aaaa.f & aaaa.g % aaaa.h ^ aaaa.i << aaaa.k >> aaaa.l**aaaa.m // aaaa.n
+):
+    return True
+if (
+    ~aaaaaaaa.a + aaaaaaaa.b - aaaaaaaa.c @ aaaaaaaa.d / aaaaaaaa.e
+    | aaaaaaaa.f & aaaaaaaa.g % aaaaaaaa.h
+    ^ aaaaaaaa.i << aaaaaaaa.k >> aaaaaaaa.l**aaaaaaaa.m // aaaaaaaa.n
+):
+    return True
+if (
+    ~aaaaaaaaaaaaaaaa.a
+    + aaaaaaaaaaaaaaaa.b
+    - aaaaaaaaaaaaaaaa.c * aaaaaaaaaaaaaaaa.d @ aaaaaaaaaaaaaaaa.e
+    | aaaaaaaaaaaaaaaa.f & aaaaaaaaaaaaaaaa.g % aaaaaaaaaaaaaaaa.h
+    ^ aaaaaaaaaaaaaaaa.i
+    << aaaaaaaaaaaaaaaa.k
+    >> aaaaaaaaaaaaaaaa.l**aaaaaaaaaaaaaaaa.m // aaaaaaaaaaaaaaaa.n
+):
+    return True
+(
+    aaaaaaaaaaaaaaaa
+    + aaaaaaaaaaaaaaaa
+    - aaaaaaaaaaaaaaaa
+    * (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)
+    / (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)
+)
```

aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa

-aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa >>

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa <<

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+(

+   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+   >> aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+   << aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+)

bbbb >> bbbb * bbbb

-aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ^bbbb.a &

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa^aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

aaaaaaaaaaaaaaaaaaaaaaa

+(

+   aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+   ^ bbbb.a & aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+   ^ aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

+)

last_call()

standalone comment at ENDMARKER

Introducing _Black_ to your project

```{note}
This guide is incomplete. Contributions are welcomed and would be deeply
appreciated!
```

Avoiding ruining git blame

A long-standing argument against moving to automated code formatters like _Black_ is
that the migration will clutter up the output of `git blame`. This was a valid argument,
but since Git version 2.23, Git natively supports
[ignoring revisions in blame](https://git-scm.com/docs/git-blame#Documentation/git-blame.txt---ignore-revltrevgt)
with the `--ignore-rev` option. You can also pass a file listing the revisions to ignore
using the `--ignore-revs-file` option. The changes made by the revision will be ignored
when assigning blame. Lines modified by an ignored revision will be blamed on the
previous revision that modified those lines.

So when migrating your project's code style to _Black_, reformat everything and commit
the changes (preferably in one massive commit). Then put the full 40 characters commit
identifier(s) into a file.

```text
# Migrate code style to Black
5b4ab991dede475d393e9d69ec388fd6bd949699
```

Afterwards, you can pass that file to `git blame` and see clean and meaningful blame

information.

```console
$ git blame important.py --ignore-revs-file .git-blame-ignore-revs
7a1ae265 (John Smith 2019-04-15 15:55:13 -0400 1) def very_important_function(text, file):
abdfd8b0 (Alice Doe  2019-09-23 11:39:32 -0400 2)     text = text.lstrip()
7a1ae265 (John Smith 2019-04-15 15:55:13 -0400 3)     with open(file, "r+") as f:
7a1ae265 (John Smith 2019-04-15 15:55:13 -0400 4)         f.write(formatted)
```

You can even configure `git` to automatically ignore revisions listed in a file on every
call to `git blame`.

```console
$ git config blame.ignoreRevsFile .git-blame-ignore-revs
```

**The one caveat is that some online Git-repositories like GitLab do not yet support
ignoring revisions using their native blame UI.** So blame information will be cluttered
with a reformatting commit on those platforms. (If you'd like this feature, there's an
open issue for [GitLab](https://gitlab.com/gitlab-org/gitlab/-/issues/31423)). This is
however supported by
[GitHub](https://docs.github.com/en/repositories/working-with-files/using-files/viewing-a-file#ignore-commits-in-
the-blame-view),
currently in beta.
[tool.black]
verbose = 1
--check = "no"
diff = "y"
color = true
line-length = 79
target-version = ["py36", "py37", "py38"]
exclude='\.pyi?$'
include='\.py?$'
python-cell-magics = ["custom1", "custom2"]

[v1.0.0-syntax]
# This shouldn't break Black.
contributors = [
 "Foo Bar <foo@example.com>",
 { name = "Baz Qux", email = "bazqux@example.com", url = "https://example.com/bazqux" }
]
Issue triage

Currently, _Black_ uses the issue tracker for bugs, feature requests, proposed design
modifications, and general user support. Each of these issues have to be triaged so they
can be eventually be resolved somehow. This document outlines the triaging process and
also the current guidelines and recommendations.

```{tip}
If you're looking for a way to contribute without submitting patches, this might be
the area for you. Since _Black_ is a popular project, its issue tracker is quite busy
and always needs more attention than is available. While triage isn't the most
glamorous or technically challenging form of contribution, it's still important.
For example, we would love to know whether that old bug report is still reproducible!

You can get easily started by reading over this document and then responding to issues.

If you contribute enough and have stayed for a long enough time, you may even be
given Triage permissions!
```

The basics

Black gets a whole bunch of different issues, they range from bug reports to user
support issues. To triage is to identify, organize, and kickstart the issue's journey
through its lifecycle to resolution.

More specifically, to triage an issue means to:

- identify what type and categories the issue falls under
- confirm bugs
- ask questions / for further information if necessary
- link related issues
- provide the first initial feedback / support

Note that triage is typically the first response to an issue, so don't fret if the issue
doesn't make much progress after initial triage. The main goal of triaging to prepare
the issue for future more specific development or discussion, so _eventually_ it will be
resolved.

The lifecycle of a bug report or user support issue typically goes something like this:

1. _the issue is waiting for triage_
2. **identified** - has been marked with a type label and other relevant labels, more
   details or a functional reproduction may be still needed (and therefore should be
   marked with `S: needs repro` or `S: awaiting response`)
3. **confirmed** - the issue can reproduced and necessary details have been provided
4. **discussion** - initial triage has been done and now the general details on how the
   issue should be best resolved are being hashed out
5. **awaiting fix** - no further discussion on the issue is necessary and a resolving PR
   is the next step
6. **closed** - the issue has been resolved, reasons include:
   - the issue couldn't be reproduced
   - the issue has been fixed
   - duplicate of another pre-existing issue or is invalid

For enhancement, documentation, and design issues, the lifecycle looks very similar but the details are different:

1. _the issue is waiting for triage_
2. **identified** - has been marked with a type label and other relevant labels
3. **discussion** - the merits of the suggested changes are currently being discussed, a
   PR would be acceptable but would be at sigificant risk of being rejected
4. **accepted & awaiting PR** - it's been determined the suggested changes are OK and a
   PR would be welcomed (`S: accepted`)
5. **closed**: - the issue has been resolved, reasons include:
   - the suggested changes were implemented
   - it was rejected (due to technical concerns, ethos conflicts, etc.)
   - duplicate of a pre-existing issue or is invalid

Note: documentation issues don't use the `S: accepted` label currently since they're less likely to be rejected.

Labelling

We use labels to organize, track progress, and help effectively divvy up work.

Our labels are divided up into several groups identified by their prefix:

- **T - Type**: the general flavor of issue / PR
- **C - Category**: areas of concerns, ranges from bug types to project maintenance
- **F - Formatting Area**: like C but for formatting specifically
- **S - Status**: what stage of resolution is this issue currently in?
- **R - Resolution**: how / why was the issue / PR resolved?

We also have a few standalone labels:

- **`good first issue`**: issues that are beginner-friendly (and will show up in GitHub
  banners for first-time visitors to the repository)
- **`help wanted`**: complex issues that need and are looking for a fair bit of work as
  to progress (will also show up in various GitHub pages)
- **`skip news`**: for PRs that are trivial and don't need a CHANGELOG entry (and skips
  the CHANGELOG entry check)

```{note}
We do use labels for PRs, in particular the `skip news` label, but we aren't that
rigorous about it. Just follow your judgement on what labels make sense for the
specific PR (if any even make sense).
```

Projects

For more general and broad goals we use projects to track work. Some may be longterm

projects with no true end (e.g. the "Amazing documentation" project) while others may be more focused and have a definite end (like the "Getting to beta" project).

```{note}
To modify GitHub Projects you need the [Write repository permission level or higher](https://docs.github.com/en/organizations/managing-access-to-your-organizations-repositories/repository-permission-levels-for-an-organization#repository-access-for-each-permission-level).
```

Closing issues

Closing an issue signifies the issue has reached the end of its life, so closing issues should be taken with care. The following is the general recommendation for each type of issue. Note that these are only guidelines and if your judgement says something else it's totally cool to go with it instead.

For most issues, closing the issue manually or automatically after a resolving PR is ideal. For bug reports specifically, if the bug has already been fixed, try to check in with the issue opener that their specific case has been resolved before closing. Note that we close issues as soon as they're fixed in the `main` branch. This doesn't necessarily mean they've been released yet.

Design and enhancement issues should be also closed when it's clear the proposed change won't be implemented, whether that has been determined after a lot of discussion or just simply goes against _Black_'s ethos. If such an issue turns heated, closing and locking is acceptable if it's severe enough (although checking in with the core team is probably a good idea).

User support issues are best closed by the author or when it's clear the issue has been resolved in some sort of manner.

Duplicates and invalid issues should always be closed since they serve no purpose and add noise to an already busy issue tracker. Although be careful to make sure it's truly a duplicate and not just very similar before labelling and closing an issue as duplicate.

Common reports

Some issues are frequently opened, like issues about _Black_ formatted code causing E203 messages. Even though these issues are probably heavily duplicated, they still require triage sucking up valuable time from other things (although they usually skip most of their lifecycle since they're closed on triage).

Here's some of the most common issues and also pre-made responses you can use:

"The trailing comma isn't being removed by Black!"

```text

Black used to remove the trailing comma if the expression fits in a single line, but this was changed by #826 and #1288. Now a trailing comma tells Black to always explode the expression. This change was made mostly for the cases where you _know_ a collection or whatever will grow in the future. Having it always exploded as one element per line reduces diff noise when adding elements. Before the "magic trailing comma" feature, you couldn't anticipate a collection's growth reliably since collections that fitted in one line were ruthlessly collapsed regardless of your intentions. One of Black's goals is reducing diff noise, so this was a good pragmatic change.

So no, this is not a bug, but an intended feature. Anyway, [here's the documentation](https://github.com/psf/black/blob/master/docs/the_black_code_style.md#the-magic-trailing-comma) on the "magic trailing comma", including the ability to skip this functionality with the `--skip-magic-trailing-comma` option. Hopefully that helps solve the possible confusion.
```

"Black formatted code is violating Flake8's E203!"

```text
Hi,

This is expected behaviour, please see the documentation regarding this case (emphasis mine):

> PEP 8 recommends to treat : in slices as a binary operator with the lowest priority, and to leave an equal amount of space on either side, **except if a parameter is omitted (e.g. ham[1 + 1 :])**. It recommends no spaces around : operators for simple expressions (ham[lower:upper]), and **extra space for complex expressions (ham[lower : upper + offset])**. **Black treats anything more than variable names as complex (ham[lower : upper + 1]).** It also states that for extended slices, both : operators have to have the same amount of spacing, except if a parameter is omitted (ham[1 + 1 ::]). Black enforces these rules consistently.

> This behaviour may raise E203 whitespace before ':' warnings in style guide enforcement tools like Flake8. **Since E203 is not PEP 8 compliant, you should tell Flake8 to ignore these warnings**.

https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html#slices

Have a good day!
```
X: int

def f(): ...

class D:
    ...

class C:
    ...

class B:

```python
    this_lack_of_newline_should_be_kept: int
    def b(self) -> None: ...

    but_this_newline_should_also_be_kept: int

class A:
    attr: int
    attr2: str

    def f(self) -> int:
        ...

    def g(self) -> str: ...



def g():
    ...

def h(): ...

if sys.version_info >= (3, 8):
    class E:
        def f(self): ...
    class F:

        def f(self): ...
    class G: ...
    class H: ...
else:
    class I: ...
    class J: ...
    def f(): ...

    class K:
        def f(self): ...
    def f(): ...

class Nested:
    class dirty: ...
    class little: ...
    class secret:
        def who_has_to_know(self): ...
    def verse(self): ...

class Conditional:
    def f(self): ...
    if sys.version_info >= (3, 8):
```

```
        def g(self): ...
    else:
        def g(self): ...
    def h(self): ...
    def i(self): ...
    if sys.version_info >= (3, 8):
        def j(self): ...
    def k(self): ...
    if sys.version_info >= (3, 8):
        class A: ...
        class B: ...
        class C:
            def l(self): ...
            def m(self): ...


# output
X: int

def f(): ...

class D: ...
class C: ...

class B:
    this_lack_of_newline_should_be_kept: int
    def b(self) -> None: ...

    but_this_newline_should_also_be_kept: int

class A:
    attr: int
    attr2: str

    def f(self) -> int: ...
    def g(self) -> str: ...

def g(): ...
def h(): ...

if sys.version_info >= (3, 8):
    class E:
        def f(self): ...

    class F:
        def f(self): ...

    class G: ...
```

```python
        class H: ...

else:
    class I: ...
    class J: ...

    def f(): ...

    class K:
        def f(self): ...

    def f(): ...

class Nested:
    class dirty: ...
    class little: ...

    class secret:
        def who_has_to_know(self): ...

    def verse(self): ...

class Conditional:
    def f(self): ...
    if sys.version_info >= (3, 8):
        def g(self): ...
    else:
        def g(self): ...

    def h(self): ...
    def i(self): ...
    if sys.version_info >= (3, 8):
        def j(self): ...

    def k(self): ...
    if sys.version_info >= (3, 8):
        class A: ...
        class B: ...

        class C:
            def l(self): ...
            def m(self): ...
```
Getting Started

New to _Black_? Don't worry, you've found the perfect place to get started!

Do you like the _Black_ code style?

Before using _Black_ on some of your code, it might be a good idea to first understand how _Black_ will format your code. _Black_ isn't for everyone and you may find something that is a dealbreaker for you personally, which is okay! The current _Black_ code style [is described here](./the_black_code_style/current_style.md).

Try it out online

Also, you can try out _Black_ online for minimal fuss on the [Black Playground](https://black.vercel.app) generously created by Jos Padilla.

Installation

Black can be installed by running `pip install black`. It requires Python 3.7+ to run. If you want to format Jupyter Notebooks, install with `pip install "black[jupyter]"`.

If you can't wait for the latest _hotness_ and want to install from GitHub, use:

`pip install git+https://github.com/psf/black`

Basic usage

To get started right away with sensible defaults:

```sh
black {source_file_or_directory}...
```

You can run _Black_ as a package if running it as a script doesn't work:

```sh
python -m black {source_file_or_directory}...
```

Next steps

Took a look at [the _Black_ code style](./the_black_code_style/current_style.md) and tried out _Black_? Fantastic, you're ready for more. Why not explore some more on using _Black_ by reading [Usage and Configuration: The basics](./usage_and_configuration/the_basics.md). Alternatively, you can check out the [Introducing _Black_ to your project](./guides/introducing_black_to_your_project.md) guide.
[project]
name = "test"
version = "1.0.0"

[tool.black]
line-length = 79

target-version = ["py310"]

[build-system]

requires = ["setuptools>=41.0", "setuptools-scm", "wheel"]

build-backend = "setuptools.build_meta"

--- [Deterministic header]

+++ [Deterministic header]

@@ -1,3 +1,3 @@

A comment-only file, with no final EOL character

This triggers https://bugs.python.org/issue2142

-# This is the line without the EOL character

\ No newline at end of file

+# This is the line without the EOL character

Change Log

Unreleased

Highlights

<!-- Include any especially major or disruptive changes here -->

Stable style

<!-- Changes that affect Black's stable style -->

Preview style

<!-- Changes that affect Black's preview style -->

Configuration

<!-- Changes to how Black can be configured -->

Packaging

<!-- Changes to how Black is packaged, such as dependency requirements -->

Parser

<!-- Changes to the parser or to version autodetection -->

Performance

<!-- Changes that improve Black's performance. -->

Output

<!-- Changes to Black's terminal output and error messages -->

Blackd

<!-- Changes to blackd -->

Integrations

<!-- For example, Docker, GitHub Actions, pre-commit, editors -->

Documentation

<!-- Major changes to documentation and policies. Small docs changes
  don't need a changelog entry. -->

23.1.0

Highlights

This is the first release of 2023, and following our
[stability policy](https://black.readthedocs.io/en/stable/the_black_code_style/index.html#stability-policy),
it comes with a number of improvements to our stable style, including improvements to
empty line handling, removal of redundant parentheses in several contexts, and output
that highlights implicitly concatenated strings better.

There are also many changes to the preview style; try out `black --preview` and give us
feedback to help us set the stable style for next year.

In addition to style changes, Black now automatically infers the supported Python
versions from your `pyproject.toml` file, removing the need to set Black's target
versions separately.

Stable style

<!-- Changes that affect Black's stable style -->

- Introduce the 2023 stable style, which incorporates most aspects of last year's
 preview style (#3418). Specific changes:
 - Enforce empty lines before classes and functions with sticky leading comments
   (#3302) (22.12.0)
 - Reformat empty and whitespace-only files as either an empty file (if no newline is
   present) or as a single newline character (if a newline is present) (#3348)
   (22.12.0)
 - Implicitly concatenated strings used as function args are now wrapped inside
   parentheses (#3307) (22.12.0)
 - Correctly handle trailing commas that are inside a line's leading non-nested parens
   (#3370) (22.12.0)
 - `--skip-string-normalization` / `-S` now prevents docstring prefixes from being
   normalized as expected (#3168) (since 22.8.0)
 - When using `--skip-magic-trailing-comma` or `-C`, trailing commas are stripped from

subscript expressions with more than 1 element (#3209) (22.8.0)
- Implicitly concatenated strings inside a list, set, or tuple are now wrapped inside parentheses (#3162) (22.8.0)
- Fix a string merging/split issue when a comment is present in the middle of implicitly concatenated strings on its own line (#3227) (22.8.0)
- Docstring quotes are no longer moved if it would violate the line length limit (#3044, #3430) (22.6.0)
- Parentheses around return annotations are now managed (#2990) (22.6.0)
- Remove unnecessary parentheses around awaited objects (#2991) (22.6.0)
- Remove unnecessary parentheses in `with` statements (#2926) (22.6.0)
- Remove trailing newlines after code block open (#3035) (22.6.0)
- Code cell separators `#%%` are now standardised to `# %%` (#2919) (22.3.0)
- Remove unnecessary parentheses from `except` statements (#2939) (22.3.0)
- Remove unnecessary parentheses from tuple unpacking in `for` loops (#2945) (22.3.0)
- Avoid magic-trailing-comma in single-element subscripts (#2942) (22.3.0)
- Fix a crash when a colon line is marked between `# fmt: off` and `# fmt: on` (#3439)

Preview style

<!-- Changes that affect Black's preview style -->

- Format hex codes in unicode escape sequences in string literals (#2916)
- Add parentheses around `if`-`else` expressions (#2278)
- Improve performance on large expressions that contain many strings (#3467)
- Fix a crash in preview style with assert + parenthesized string (#3415)
- Fix crashes in preview style with walrus operators used in function return annotations and except clauses (#3423)
- Fix a crash in preview advanced string processing where mixed implicitly concatenated regular and f-strings start with an empty span (#3463)
- Fix a crash in preview advanced string processing where a standalone comment is placed before a dict's value (#3469)
- Fix an issue where extra empty lines are added when a decorator has `# fmt: skip` applied or there is a standalone comment between decorators (#3470)
- Do not put the closing quotes in a docstring on a separate line, even if the line is too long (#3430)
- Long values in dict literals are now wrapped in parentheses; correspondingly unnecessary parentheses around short values in dict literals are now removed; long string lambda values are now wrapped in parentheses (#3440)
- Fix two crashes in preview style involving edge cases with docstrings (#3451)
- Exclude string type annotations from improved string processing; fix crash when the return type annotation is stringified and spans across multiple lines (#3462)
- Wrap multiple context managers in parentheses when targeting Python 3.9+ (#3489)
- Fix several crashes in preview style with walrus operators used in `with` statements or tuples (#3473)
- Fix an invalid quote escaping bug in f-string expressions where it produced invalid code. Implicitly concatenated f-strings with different quotes can now be merged or quote-normalized by changing the quotes used in expressions. (#3509)
- Fix crash on `await (yield)` when Black is compiled with mypyc (#3533)

Configuration

<!-- Changes to how Black can be configured -->

- Black now tries to infer its `--target-version` from the project metadata specified in
 `pyproject.toml` (#3219)

Packaging

<!-- Changes to how Black is packaged, such as dependency requirements -->

- Upgrade mypyc from `0.971` to `0.991` so mypycified _Black_ can be built on armv7
 (#3380)
 - This also fixes some crashes while using compiled Black with a debug build of
   CPython
- Drop specific support for the `tomli` requirement on 3.11 alpha releases, working
 around a bug that would cause the requirement not to be installed on any non-final
 Python releases (#3448)
- Black now depends on `packaging` version `22.0` or later. This is required for new
 functionality that needs to parse part of the project metadata (#3219)

Output

<!-- Changes to Black's terminal output and error messages -->

- Calling `black --help` multiple times will return the same help contents each time
 (#3516)
- Verbose logging now shows the values of `pyproject.toml` configuration variables
 (#3392)
- Fix false symlink detection messages in verbose output due to using an incorrect
 relative path to the project root (#3385)

Integrations

<!-- For example, Docker, GitHub Actions, pre-commit, editors -->

- Move 3.11 CI to normal flow now that all dependencies support 3.11 (#3446)
- Docker: Add new `latest_prerelease` tag automation to follow latest black alpha
 release on docker images (#3465)

Documentation

<!-- Major changes to documentation and policies. Small docs changes
    don't need a changelog entry. -->

- Expand `vim-plug` installation instructions to offer more explicit options (#3468)

22.12.0

Preview style

<!-- Changes that affect Black's preview style -->

- Enforce empty lines before classes and functions with sticky leading comments (#3302)
- Reformat empty and whitespace-only files as either an empty file (if no newline is
  present) or as a single newline character (if a newline is present) (#3348)
- Implicitly concatenated strings used as function args are now wrapped inside
  parentheses (#3307)
- For assignment statements, prefer splitting the right hand side if the left hand side
  fits on a single line (#3368)
- Correctly handle trailing commas that are inside a line's leading non-nested parens
  (#3370)

Configuration

<!-- Changes to how Black can be configured -->

- Fix incorrectly applied `.gitignore` rules by considering the `.gitignore` location
  and the relative path to the target file (#3338)
- Fix incorrectly ignoring `.gitignore` presence when more than one source directory is
  specified (#3336)

Parser

<!-- Changes to the parser or to version autodetection -->

- Parsing support has been added for walruses inside generator expression that are
  passed as function args (for example,
  `any(match := my_re.match(text) for text in texts)`) (#3327).

Integrations

<!-- For example, Docker, GitHub Actions, pre-commit, editors -->

- Vim plugin: Optionally allow using the system installation of Black via
  `let g:black_use_virtualenv = 0`(#3309)

22.10.0

Highlights

- Runtime support for Python 3.6 has been removed. Formatting 3.6 code will still be
  supported until further notice.

Stable style

- Fix a crash when `# fmt: on` is used on a different block level than `# fmt: off`
 (#3281)

Preview style

- Fix a crash when formatting some dicts with parenthesis-wrapped long string keys
 (#3262)

Configuration

- `.ipynb_checkpoints` directories are now excluded by default (#3293)
- Add `--skip-source-first-line` / `-x` option to ignore the first line of source code
 while formatting (#3299)

Packaging

- Executables made with PyInstaller will no longer crash when formatting several files
 at once on macOS. Native x86-64 executables for macOS are available once again.
 (#3275)
- Hatchling is now used as the build backend. This will not have any effect for users
 who install Black with its wheels from PyPI. (#3233)
- Faster compiled wheels are now available for CPython 3.11 (#3276)

Blackd

- Windows style (CRLF) newlines will be preserved (#3257).

Integrations

- Vim plugin: add flag (`g:black_preview`) to enable/disable the preview style (#3246)
- Update GitHub Action to support formatting of Jupyter Notebook files via a `jupyter`
 option (#3282)
- Update GitHub Action to support use of version specifiers (e.g. `<23`) for Black
 version (#3265)

22.8.0

Highlights

- Python 3.11 is now supported, except for _blackd_ as aiohttp does not support 3.11 as
 of publishing (#3234)
- This is the last release that supports running _Black_ on Python 3.6 (formatting 3.6
 code will continue to be supported until further notice)
- Reword the stability policy to say that we may, in rare cases, make changes that
 affect code that was not previously formatted by _Black_ (#3155)

Stable style

- Fix an infinite loop when using `# fmt: on/off` in the middle of an expression or code block (#3158)
- Fix incorrect handling of `# fmt: skip` on colon (`:`) lines (#3148)
- Comments are no longer deleted when a line had spaces removed around power operators (#2874)

Preview style

- Single-character closing docstring quotes are no longer moved to their own line as this is invalid. This was a bug introduced in version 22.6.0. (#3166)
- `--skip-string-normalization` / `-S` now prevents docstring prefixes from being normalized as expected (#3168)
- When using `--skip-magic-trailing-comma` or `-C`, trailing commas are stripped from subscript expressions with more than 1 element (#3209)
- Implicitly concatenated strings inside a list, set, or tuple are now wrapped inside parentheses (#3162)
- Fix a string merging/split issue when a comment is present in the middle of implicitly concatenated strings on its own line (#3227)

Blackd

- `blackd` now supports enabling the preview style via the `X-Preview` header (#3217)

Configuration

- Black now uses the presence of debug f-strings to detect target version (#3215)
- Fix misdetection of project root and verbose logging of sources in cases involving `--stdin-filename` (#3216)
- Immediate `.gitignore` files in source directories given on the command line are now also respected, previously only `.gitignore` files in the project root and automatically discovered directories were respected (#3237)

Documentation

- Recommend using BlackConnect in IntelliJ IDEs (#3150)

Integrations

- Vim plugin: prefix messages with `Black: ` so it's clear they come from Black (#3194)
- Docker: changed to a /opt/venv installation + added to PATH to be available to non-root users (#3202)

Output

- Change from deprecated `asyncio.get_event_loop()` to create our event loop which removes DeprecationWarning (#3164)
- Remove logging from internal `blib2to3` library since it regularly emits error logs

about failed caching that can and should be ignored (#3193)

Parser

- Type comments are now included in the AST equivalence check consistently so accidental deletion raises an error. Though type comments can't be tracked when running on PyPy 3.7 due to standard library limitations. (#2874)

Performance

- Reduce Black's startup time when formatting a single file by 15-30% (#3211)

22.6.0

Style

- Fix unstable formatting involving `#fmt: skip` and `# fmt:skip` comments (notice the lack of spaces) (#2970)

Preview style

- Docstring quotes are no longer moved if it would violate the line length limit (#3044)
- Parentheses around return annotations are now managed (#2990)
- Remove unnecessary parentheses around awaited objects (#2991)
- Remove unnecessary parentheses in `with` statements (#2926)
- Remove trailing newlines after code block open (#3035)

Integrations

- Add `scripts/migrate-black.py` script to ease introduction of Black to a Git project (#3038)

Output

- Output Python version and implementation as part of `--version` flag (#2997)

Packaging

- Use `tomli` instead of `tomllib` on Python 3.11 builds where `tomllib` is not available (#2987)

Parser

- [PEP 654](https://peps.python.org/pep-0654/#except) syntax (for example, `except *ExceptionGroup:`) is now supported (#3016)
- [PEP 646](https://peps.python.org/pep-0646) syntax (for example, `Array[Batch, *Shape]` or `def fn(*args: *T) -> None`) is now supported (#3071)

Vim Plugin

- Fix `strtobool` function. It didn't parse true/on/false/off. (#3025)

22.3.0

Preview style

- Code cell separators `#%%` are now standardised to `# %%` (#2919)
- Remove unnecessary parentheses from `except` statements (#2939)
- Remove unnecessary parentheses from tuple unpacking in `for` loops (#2945)
- Avoid magic-trailing-comma in single-element subscripts (#2942)

Configuration

- Do not format `__pypackages__` directories by default (#2836)
- Add support for specifying stable version with `--required-version` (#2832).
- Avoid crashing when the user has no homedir (#2814)
- Avoid crashing when md5 is not available (#2905)
- Fix handling of directory junctions on Windows (#2904)

Documentation

- Update pylint config documentation (#2931)

Integrations

- Move test to disable plugin in Vim/Neovim, which speeds up loading (#2896)

Output

- In verbose mode, log when _Black_ is using user-level config (#2861)

Packaging

- Fix Black to work with Click 8.1.0 (#2966)
- On Python 3.11 and newer, use the standard library's `tomllib` instead of `tomli`
  (#2903)
- `black-primer`, the deprecated internal devtool, has been removed and copied to a
  [separate repository](https://github.com/cooperlees/black-primer) (#2924)

Parser

- Black can now parse starred expressions in the target of `for` and `async for`
  statements, e.g `for item in *items_1, *items_2: pass` (#2879).

22.1.0

At long last, _Black_ is no longer a beta product! This is the first non-beta release
and the first release covered by our new
[stability policy](https://black.readthedocs.io/en/stable/the_black_code_style/index.html#stability-policy).

Highlights

- **Remove Python 2 support** (#2740)
- Introduce the `--preview` flag (#2752)

Style

- Deprecate `--experimental-string-processing` and move the functionality under
  `--preview` (#2789)
- For stubs, one blank line between class attributes and methods is now kept if there's
  at least one pre-existing blank line (#2736)
- Black now normalizes string prefix order (#2297)
- Remove spaces around power operators if both operands are simple (#2726)
- Work around bug that causes unstable formatting in some cases in the presence of the
  magic trailing comma (#2807)
- Use parentheses for attribute access on decimal float and int literals (#2799)
- Don't add whitespace for attribute access on hexadecimal, binary, octal, and complex
  literals (#2799)
- Treat blank lines in stubs the same inside top-level `if` statements (#2820)
- Fix unstable formatting with semicolons and arithmetic expressions (#2817)
- Fix unstable formatting around magic trailing comma (#2572)

Parser

- Fix mapping cases that contain as-expressions, like `case {"key": 1 | 2 as password}`
  (#2686)
- Fix cases that contain multiple top-level as-expressions, like `case 1 as a, 2 as b`
  (#2716)
- Fix call patterns that contain as-expressions with keyword arguments, like
  `case Foo(bar=baz as quux)` (#2749)
- Tuple unpacking on `return` and `yield` constructs now implies 3.8+ (#2700)
- Unparenthesized tuples on annotated assignments (e.g
  `values: Tuple[int, ...] = 1, 2, 3`) now implies 3.8+ (#2708)
- Fix handling of standalone `match()` or `case()` when there is a trailing newline or a
  comment inside of the parentheses. (#2760)
- `from __future__ import annotations` statement now implies Python 3.7+ (#2690)

Performance

- Speed-up the new backtracking parser about 4X in general (enabled when
  `--target-version` is set to 3.10 and higher). (#2728)
- _Black_ is now compiled with [mypyc](https://github.com/mypyc/mypyc) for an overall 2x
  speed-up. 64-bit Windows, MacOS, and Linux (not including musl) are supported. (#1009,
  #2431)

Configuration

- Do not accept bare carriage return line endings in pyproject.toml (#2408)
- Add configuration option (`python-cell-magics`) to format cells with custom magics in Jupyter Notebooks (#2744)
- Allow setting custom cache directory on all platforms with environment variable `BLACK_CACHE_DIR` (#2739).
- Enable Python 3.10+ by default, without any extra need to specify `--target-version=py310`. (#2758)
- Make passing `SRC` or `--code` mandatory and mutually exclusive (#2804)

Output

- Improve error message for invalid regular expression (#2678)
- Improve error message when parsing fails during AST safety check by embedding the underlying SyntaxError (#2693)
- No longer color diff headers white as it's unreadable in light themed terminals (#2691)
- Text coloring added in the final statistics (#2712)
- Verbose mode also now describes how a project root was discovered and which paths will be formatted. (#2526)

Packaging

- All upper version bounds on dependencies have been removed (#2718)
- `typing-extensions` is no longer a required dependency in Python 3.10+ (#2772)
- Set `click` lower bound to `8.0.0` (#2791)

Integrations

- Update GitHub action to support containerized runs (#2748)

Documentation

- Change protocol in pip installation instructions to `https://` (#2761)
- Change HTML theme to Furo primarily for its responsive design and mobile support (#2793)
- Deprecate the `black-primer` tool (#2809)
- Document Python support policy (#2819)

21.12b0

Black

- Fix determination of f-string expression spans (#2654)
- Fix bad formatting of error messages about EOF in multi-line statements (#2343)
- Functions and classes in blocks now have more consistent surrounding spacing (#2472)

Jupyter Notebook support

- Cell magics are now only processed if they are known Python cell magics. Earlier, all
  cell magics were tokenized, leading to possible indentation errors e.g. with
  `%%writefile`. (#2630)
- Fix assignment to environment variables in Jupyter Notebooks (#2642)

Python 3.10 support

- Point users to using `--target-version py310` if we detect 3.10-only syntax (#2668)
- Fix `match` statements with open sequence subjects, like `match a, b:` or
  `match a, *b:` (#2639) (#2659)
- Fix `match`/`case` statements that contain `match`/`case` soft keywords multiple
  times, like `match re.match()` (#2661)
- Fix `case` statements with an inline body (#2665)
- Fix styling of starred expressions inside `match` subject (#2667)
- Fix parser error location on invalid syntax in a `match` statement (#2649)
- Fix Python 3.10 support on platforms without ProcessPoolExecutor (#2631)
- Improve parsing performance on code that uses `match` under `--target-version py310`
  up to ~50% (#2670)

Packaging

- Remove dependency on `regex` (#2644) (#2663)

21.11b1

Black

- Bumped regex version minimum to 2021.4.4 to fix Pattern class usage (#2621)

21.11b0

Black

- Warn about Python 2 deprecation in more cases by improving Python 2 only syntax
  detection (#2592)
- Add experimental PyPy support (#2559)
- Add partial support for the match statement. As it's experimental, it's only enabled
  when `--target-version py310` is explicitly specified (#2586)
- Add support for parenthesized with (#2586)
- Declare support for Python 3.10 for running Black (#2562)

Integrations

- Fixed vim plugin with Python 3.10 by removing deprecated distutils import (#2610)
- The vim plugin now parses `skip_magic_trailing_comma` from pyproject.toml (#2613)

21.10b0

Black

- Document stability policy, that will apply for non-beta releases (#2529)
- Add new `--workers` parameter (#2514)
- Fixed feature detection for positional-only arguments in lambdas (#2532)
- Bumped typed-ast version minimum to 1.4.3 for 3.10 compatibility (#2519)
- Fixed a Python 3.10 compatibility issue where the loop argument was still being passed
 even though it has been removed (#2580)
- Deprecate Python 2 formatting support (#2523)

Blackd

- Remove dependency on aiohttp-cors (#2500)
- Bump required aiohttp version to 3.7.4 (#2509)

Black-Primer

- Add primer support for --projects (#2555)
- Print primer summary after individual failures (#2570)

Integrations

- Allow to pass `target_version` in the vim plugin (#1319)
- Install build tools in docker file and use multi-stage build to keep the image size
 down (#2582)

21.9b0

Packaging

- Fix missing modules in self-contained binaries (#2466)
- Fix missing toml extra used during installation (#2475)

21.8b0

Black

- Add support for formatting Jupyter Notebook files (#2357)
- Move from `appdirs` dependency to `platformdirs` (#2375)
- Present a more user-friendly error if .gitignore is invalid (#2414)
- The failsafe for accidentally added backslashes in f-string expressions has been
 hardened to handle more edge cases during quote normalization (#2437)
- Avoid changing a function return type annotation's type to a tuple by adding a
 trailing comma (#2384)
- Parsing support has been added for unparenthesized walruses in set literals, set

comprehensions, and indices (#2447).
- Pin `setuptools-scm` build-time dependency version (#2457)
- Exclude typing-extensions version 3.10.0.1 due to it being broken on Python 3.10
 (#2460)

Blackd

- Replace sys.exit(-1) with raise ImportError as it plays more nicely with tools that
 scan installed packages (#2440)

Integrations

- The provided pre-commit hooks no longer specify `language_version` to avoid overriding
 `default_language_version` (#2430)

21.7b0

Black

- Configuration files using TOML features higher than spec v0.5.0 are now supported
 (#2301)
- Add primer support and test for code piped into black via STDIN (#2315)
- Fix internal error when `FORCE_OPTIONAL_PARENTHESES` feature is enabled (#2332)
- Accept empty stdin (#2346)
- Provide a more useful error when parsing fails during AST safety checks (#2304)

Docker

- Add new `latest_release` tag automation to follow latest black release on docker
 images (#2374)

Integrations

- The vim plugin now searches upwards from the directory containing the current buffer
 instead of the current working directory for pyproject.toml. (#1871)
- The vim plugin now reads the correct string normalization option in pyproject.toml
 (#1869)
- The vim plugin no longer crashes Black when there's boolean values in pyproject.toml
 (#1869)

21.6b0

Black

- Fix failure caused by `fmt: skip` and indentation (#2281)
- Account for += assignment when deciding whether to split string (#2312)
- Correct max string length calculation when there are string operators (#2292)
- Fixed option usage when using the `--code` flag (#2259)

- Do not call `uvloop.install()` when _Black_ is used as a library (#2303)
- Added `--required-version` option to require a specific version to be running (#2300)
- Fix incorrect custom breakpoint indices when string group contains fake f-strings (#2311)
- Fix regression where `R` prefixes would be lowercased for docstrings (#2285)
- Fix handling of named escapes (`\N{...}`) when `--experimental-string-processing` is used (#2319)

Integrations

- The official Black action now supports choosing what version to use, and supports the major 3 OSes. (#1940)

21.5b2

Black

- A space is no longer inserted into empty docstrings (#2249)
- Fix handling of .gitignore files containing non-ASCII characters on Windows (#2229)
- Respect `.gitignore` files in all levels, not only `root/.gitignore` file (apply `.gitignore` rules like `git` does) (#2225)
- Restored compatibility with Click 8.0 on Python 3.6 when LANG=C used (#2227)
- Add extra uvloop install + import support if in python env (#2258)
- Fix --experimental-string-processing crash when matching parens are not found (#2283)
- Make sure to split lines that start with a string operator (#2286)
- Fix regular expression that black uses to identify f-expressions (#2287)

Blackd

- Add a lower bound for the `aiohttp-cors` dependency. Only 0.4.0 or higher is supported. (#2231)

Packaging

- Release self-contained x86_64 MacOS binaries as part of the GitHub release pipeline (#2198)
- Always build binaries with the latest available Python (#2260)

Documentation

- Add discussion of magic comments to FAQ page (#2272)
- `--experimental-string-processing` will be enabled by default in the future (#2273)
- Fix typos discovered by codespell (#2228)
- Fix Vim plugin installation instructions. (#2235)
- Add new Frequently Asked Questions page (#2247)
- Fix encoding + symlink issues preventing proper build on Windows (#2262)

21.5b1

Black

- Refactor `src/black/__init__.py` into many files (#2206)

Documentation

- Replaced all remaining references to the
 [`master`](https://github.com/psf/black/tree/main) branch with the
 [`main`](https://github.com/psf/black/tree/main) branch. Some additional changes in
 the source code were also made. (#2210)
- Sigificantly reorganized the documentation to make much more sense. Check them out by
 heading over to [the stable docs on RTD](https://black.readthedocs.io/en/stable/).
 (#2174)

21.5b0

Black

- Set `--pyi` mode if `--stdin-filename` ends in `.pyi` (#2169)
- Stop detecting target version as Python 3.9+ with pre-PEP-614 decorators that are
 being called but with no arguments (#2182)

Black-Primer

- Add `--no-diff` to black-primer to suppress formatting changes (#2187)

21.4b2

Black

- Fix crash if the user configuration directory is inaccessible. (#2158)

- Clarify
 [circumstances](https://github.com/psf/black/blob/master/docs/the_black_code_style.md#pragmatism)
 in which _Black_ may change the AST (#2159)

- Allow `.gitignore` rules to be overridden by specifying `exclude` in `pyproject.toml`
 or on the command line. (#2170)

Packaging

- Install `primer.json` (used by `black-primer` by default) with black. (#2154)

21.4b1

Black

- Fix crash on docstrings ending with "\\ ". (#2142)

- Fix crash when atypical whitespace is cleaned out of dostrings (#2120)

- Reflect the `--skip-magic-trailing-comma` and `--experimental-string-processing` flags
  in the name of the cache file. Without this fix, changes in these flags would not take
  effect if the cache had already been populated. (#2131)

- Don't remove necessary parentheses from assignment expression containing assert /
  return statements. (#2143)

Packaging

- Bump pathspec to >= 0.8.1 to solve invalid .gitignore exclusion handling

21.4b0

Black

- Fixed a rare but annoying formatting instability created by the combination of
  optional trailing commas inserted by `Black` and optional parentheses looking at
  pre-existing "magic" trailing commas. This fixes issue #1629 and all of its many many
  duplicates. (#2126)

- `Black` now processes one-line docstrings by stripping leading and trailing spaces,
  and adding a padding space when needed to break up """". (#1740)

- `Black` now cleans up leading non-breaking spaces in comments (#2092)

- `Black` now respects `--skip-string-normalization` when normalizing multiline
  docstring quotes (#1637)

- `Black` no longer removes all empty lines between non-function code and decorators
  when formatting typing stubs. Now `Black` enforces a single empty line. (#1646)

- `Black` no longer adds an incorrect space after a parenthesized assignment expression
  in if/while statements (#1655)

- Added `--skip-magic-trailing-comma` / `-C` to avoid using trailing commas as a reason
  to split lines (#1824)

- fixed a crash when PWD=/ on POSIX (#1631)

- fixed "I/O operation on closed file" when using --diff (#1664)

- Prevent coloured diff output being interleaved with multiple files (#1673)

- Added support for PEP 614 relaxed decorator syntax on python 3.9 (#1711)

- Added parsing support for unparenthesized tuples and yield expressions in annotated assignments (#1835)

- added `--extend-exclude` argument (PR #2005)

- speed up caching by avoiding pathlib (#1950)

- `--diff` correctly indicates when a file doesn't end in a newline (#1662)

- Added `--stdin-filename` argument to allow stdin to respect `--force-exclude` rules (#1780)

- Lines ending with `fmt: skip` will now be not formatted (#1800)

- PR #2053: Black no longer relies on typed-ast for Python 3.8 and higher

- PR #2053: Python 2 support is now optional, install with `python3 -m pip install black[python2]` to maintain support.

- Exclude `venv` directory by default (#1683)

- Fixed "Black produced code that is not equivalent to the source" when formatting Python 2 docstrings (#2037)

Packaging

- Self-contained native _Black_ binaries are now provided for releases via GitHub Releases (#1743)

20.8b1

Packaging

- explicitly depend on Click 7.1.2 or newer as `Black` no longer works with versions older than 7.0

20.8b0

Black

- re-implemented support for explicit trailing commas: now it works consistently within any bracket pair, including nested structures (#1288 and duplicates)

- `Black` now reindents docstrings when reindenting code around it (#1053)

- `Black` now shows colored diffs (#1266)

- `Black` is now packaged using 'py3' tagged wheels (#1388)

- `Black` now supports Python 3.8 code, e.g. star expressions in return statements (#1121)

- `Black` no longer normalizes capital R-string prefixes as those have a community-accepted meaning (#1244)

- `Black` now uses exit code 2 when specified configuration file doesn't exit (#1361)

- `Black` now works on AWS Lambda (#1141)

- added `--force-exclude` argument (#1032)

- removed deprecated `--py36` option (#1236)

- fixed `--diff` output when EOF is encountered (#526)

- fixed `# fmt: off` handling around decorators (#560)

- fixed unstable formatting with some `# type: ignore` comments (#1113)

- fixed invalid removal on organizing brackets followed by indexing (#1575)

- introduced `black-primer`, a CI tool that allows us to run regression tests against existing open source users of Black (#1402)

- introduced property-based fuzzing to our test suite based on Hypothesis and Hypothersmith (#1566)

- implemented experimental and disabled by default long string rewrapping (#1132), hidden under a `--experimental-string-processing` flag while it's being worked on; this is an undocumented and unsupported feature, you lose Internet points for depending on it (#1609)

Vim plugin

- prefer virtualenv packages over global packages (#1383)

19.10b0

- added support for PEP 572 assignment expressions (#711)

- added support for PEP 570 positional-only arguments (#943)

- added support for async generators (#593)

- added support for pre-splitting collections by putting an explicit trailing comma

inside (#826)

- added `black -c` as a way to format code passed from the command line (#761)

- --safe now works with Python 2 code (#840)

- fixed grammar selection for Python 2-specific code (#765)

- fixed feature detection for trailing commas in function definitions and call sites (#763)

- `# fmt: off`/`# fmt: on` comment pairs placed multiple times within the same block of code now behave correctly (#1005)

- _Black_ no longer crashes on Windows machines with more than 61 cores (#838)

- _Black_ no longer crashes on standalone comments prepended with a backslash (#767)

- _Black_ no longer crashes on `from` ... `import` blocks with comments (#829)

- _Black_ no longer crashes on Python 3.7 on some platform configurations (#494)

- _Black_ no longer fails on comments in from-imports (#671)

- _Black_ no longer fails when the file starts with a backslash (#922)

- _Black_ no longer merges regular comments with type comments (#1027)

- _Black_ no longer splits long lines that contain type comments (#997)

- removed unnecessary parentheses around `yield` expressions (#834)

- added parentheses around long tuples in unpacking assignments (#832)

- added parentheses around complex powers when they are prefixed by a unary operator (#646)

- fixed bug that led _Black_ format some code with a line length target of 1 (#762)

- _Black_ no longer introduces quotes in f-string subexpressions on string boundaries (#863)

- if _Black_ puts parenthesis around a single expression, it moves comments to the wrapped expression instead of after the brackets (#872)

- `blackd` now returns the version of _Black_ in the response headers (#1013)

- `blackd` can now output the diff of formats on source code when the `X-Diff` header is

provided (#969)

19.3b0

- new option `--target-version` to control which Python versions _Black_-formatted code
 should target (#618)

- deprecated `--py36` (use `--target-version=py36` instead) (#724)

- _Black_ no longer normalizes numeric literals to include `_` separators (#696)

- long `del` statements are now split into multiple lines (#698)

- type comments are no longer mangled in function signatures

- improved performance of formatting deeply nested data structures (#509)

- _Black_ now properly formats multiple files in parallel on Windows (#632)

- _Black_ now creates cache files atomically which allows it to be used in parallel
 pipelines (like `xargs -P8`) (#673)

- _Black_ now correctly indents comments in files that were previously formatted with
 tabs (#262)

- `blackd` now supports CORS (#622)

18.9b0

- numeric literals are now formatted by _Black_ (#452, #461, #464, #469):

 - numeric literals are normalized to include `_` separators on Python 3.6+ code

 - added `--skip-numeric-underscore-normalization` to disable the above behavior and
   leave numeric underscores as they were in the input

 - code with `_` in numeric literals is recognized as Python 3.6+

 - most letters in numeric literals are lowercased (e.g., in `1e10`, `0x01`)

 - hexadecimal digits are always uppercased (e.g. `0xBADC0DE`)

- added `blackd`, see
 [its documentation](https://github.com/psf/black/blob/18.9b0/README.md#blackd) for
 more info (#349)

- adjacent string literals are now correctly split into multiple lines (#463)

- trailing comma is now added to single imports that don't fit on a line (#250)

- cache is now populated when `--check` is successful for a file which speeds up
 consecutive checks of properly formatted unmodified files (#448)

- whitespace at the beginning of the file is now removed (#399)

- fixed mangling [pweave](http://mpastell.com/pweave/) and
 [Spyder IDE](https://www.spyder-ide.org/) special comments (#532)

- fixed unstable formatting when unpacking big tuples (#267)

- fixed parsing of `__future__` imports with renames (#389)

- fixed scope of `# fmt: off` when directly preceding `yield` and other nodes (#385)

- fixed formatting of lambda expressions with default arguments (#468)

- fixed `async for` statements: _Black_ no longer breaks them into separate lines (#372)

- note: the Vim plugin stopped registering `,=` as a default chord as it turned out to
 be a bad idea (#415)

18.6b4

- hotfix: don't freeze when multiple comments directly precede `# fmt: off` (#371)

18.6b3

- typing stub files (`.pyi`) now have blank lines added after constants (#340)

- `# fmt: off` and `# fmt: on` are now much more dependable:

 - they now work also within bracket pairs (#329)

 - they now correctly work across function/class boundaries (#335)

 - they now work when an indentation block starts with empty lines or misaligned
   comments (#334)

- made Click not fail on invalid environments; note that Click is right but the
 likelihood we'll need to access non-ASCII file paths when dealing with Python source
 code is low (#277)

- fixed improper formatting of f-strings with quotes inside interpolated expressions
 (#322)

- fixed unnecessary slowdown when long list literals where found in a file

- fixed unnecessary slowdown on AST nodes with very many siblings

- fixed cannibalizing backslashes during string normalization

- fixed a crash due to symbolic links pointing outside of the project directory (#338)

18.6b2

- added `--config` (#65)

- added `-h` equivalent to `--help` (#316)

- fixed improper unmodified file caching when `-S` was used

- fixed extra space in string unpacking (#305)

- fixed formatting of empty triple quoted strings (#313)

- fixed unnecessary slowdown in comment placement calculation on lines without comments

18.6b1

- hotfix: don't output human-facing information on stdout (#299)

- hotfix: don't output cake emoji on non-zero return code (#300)

18.6b0

- added `--include` and `--exclude` (#270)

- added `--skip-string-normalization` (#118)

- added `--verbose` (#283)

- the header output in `--diff` now actually conforms to the unified diff spec

- fixed long trivial assignments being wrapped in unnecessary parentheses (#273)

- fixed unnecessary parentheses when a line contained multiline strings (#232)

- fixed stdin handling not working correctly if an old version of Click was used (#276)

- _Black_ now preserves line endings when formatting a file in place (#258)

18.5b1

- added `--pyi` (#249)

- added `--py36` (#249)

- Python grammar pickle caches are stored with the formatting caches, making _Black_ work in environments where site-packages is not user-writable (#192)

- _Black_ now enforces a PEP 257 empty line after a class-level docstring (and/or fields) and the first method

- fixed invalid code produced when standalone comments were present in a trailer that was omitted from line splitting on a large expression (#237)

- fixed optional parentheses being removed within `# fmt: off` sections (#224)

- fixed invalid code produced when stars in very long imports were incorrectly wrapped in optional parentheses (#234)

- fixed unstable formatting when inline comments were moved around in a trailer that was omitted from line splitting on a large expression (#238)

- fixed extra empty line between a class declaration and the first method if no class docstring or fields are present (#219)

- fixed extra empty line between a function signature and an inner function or inner class (#196)

18.5b0

- call chains are now formatted according to the [fluent interfaces](https://en.wikipedia.org/wiki/Fluent_interface) style (#67)

- data structure literals (tuples, lists, dictionaries, and sets) are now also always exploded like imports when they don't fit in a single line (#152)

- slices are now formatted according to PEP 8 (#178)

- parentheses are now also managed automatically on the right-hand side of assignments and return statements (#140)

- math operators now use their respective priorities for delimiting multiline expressions (#148)

- optional parentheses are now omitted on expressions that start or end with a bracket and only contain a single operator (#177)

- empty parentheses in a class definition are now removed (#145, #180)

- string prefixes are now standardized to lowercase and `u` is removed on Python 3.6+

only code and Python 2.7+ code with the `unicode_literals` future import (#188, #198, #199)

- typing stub files (`.pyi`) are now formatted in a style that is consistent with PEP 484 (#207, #210)

- progress when reformatting many files is now reported incrementally

- fixed trailers (content with brackets) being unnecessarily exploded into their own lines after a dedented closing bracket (#119)

- fixed an invalid trailing comma sometimes left in imports (#185)

- fixed non-deterministic formatting when multiple pairs of removable parentheses were used (#183)

- fixed multiline strings being unnecessarily wrapped in optional parentheses in long assignments (#215)

- fixed not splitting long from-imports with only a single name

- fixed Python 3.6+ file discovery by also looking at function calls with unpacking. This fixed non-deterministic formatting if trailing commas where used both in function signatures with stars and function calls with stars but the former would be reformatted to a single line.

- fixed crash on dealing with optional parentheses (#193)

- fixed "is", "is not", "in", and "not in" not considered operators for splitting purposes

- fixed crash when dead symlinks where encountered

18.4a4

- don't populate the cache on `--check` (#175)

18.4a3

- added a "cache"; files already reformatted that haven't changed on disk won't be reformatted again (#109)

- `--check` and `--diff` are no longer mutually exclusive (#149)

- generalized star expression handling, including double stars; this fixes multiplication making expressions "unsafe" for trailing commas (#132)

- _Black_ no longer enforces putting empty lines behind control flow statements (#90)

- _Black_ now splits imports like "Mode 3 + trailing comma" of isort (#127)

- fixed comment indentation when a standalone comment closes a block (#16, #32)

- fixed standalone comments receiving extra empty lines if immediately preceding a
  class, def, or decorator (#56, #154)

- fixed `--diff` not showing entire path (#130)

- fixed parsing of complex expressions after star and double stars in function calls
  (#2)

- fixed invalid splitting on comma in lambda arguments (#133)

- fixed missing splits of ternary expressions (#141)

18.4a2

- fixed parsing of unaligned standalone comments (#99, #112)

- fixed placement of dictionary unpacking inside dictionary literals (#111)

- Vim plugin now works on Windows, too

- fixed unstable formatting when encountering unnecessarily escaped quotes in a string
  (#120)

18.4a1

- added `--quiet` (#78)

- added automatic parentheses management (#4)

- added [pre-commit](https://pre-commit.com) integration (#103, #104)

- fixed reporting on `--check` with multiple files (#101, #102)

- fixed removing backslash escapes from raw strings (#100, #105)

18.4a0

- added `--diff` (#87)

- add line breaks before all delimiters, except in cases like commas, to better comply
  with PEP 8 (#73)

- standardize string literals to use double quotes (almost) everywhere (#75)

- fixed handling of standalone comments within nested bracketed expressions; _Black_ will no longer produce super long lines or put all standalone comments at the end of the expression (#22)

- fixed 18.3a4 regression: don't crash and burn on empty lines with trailing whitespace (#80)

- fixed 18.3a4 regression: `# yapf: disable` usage as trailing comment would cause _Black_ to not emit the rest of the file (#95)

- when CTRL+C is pressed while formatting many files, _Black_ no longer freaks out with a flurry of asyncio-related exceptions

- only allow up to two empty lines on module level and only single empty lines within functions (#74)

18.3a4

- `# fmt: off` and `# fmt: on` are implemented (#5)

- automatic detection of deprecated Python 2 forms of print statements and exec statements in the formatted file (#49)

- use proper spaces for complex expressions in default values of typed function arguments (#60)

- only return exit code 1 when --check is used (#50)

- don't remove single trailing commas from square bracket indexing (#59)

- don't omit whitespace if the previous factor leaf wasn't a math operator (#55)

- omit extra space in kwarg unpacking if it's the first argument (#46)

- omit extra space in
  [Sphinx auto-attribute comments](http://www.sphinx-doc.org/en/stable/ext/autodoc.html#directive-autoattribute)
  (#68)

18.3a3

- don't remove single empty lines outside of bracketed expressions (#19)

- added ability to pipe formatting from stdin to stdin (#25)

- restored ability to format code with legacy usage of `async` as a name (#20, #42)

- even better handling of numpy-style array indexing (#33, again)

18.3a2

- changed positioning of binary operators to occur at beginning of lines instead of at
 the end, following
 [a recent change to PEP
8](https://github.com/python/peps/commit/c59c4376ad233a62ca4b3a6060c81368bd21e85b)
 (#21)

- ignore empty bracket pairs while splitting. This avoids very weirdly looking
 formattings (#34, #35)

- remove a trailing comma if there is a single argument to a call

- if top level functions were separated by a comment, don't put four empty lines after
 the upper function

- fixed unstable formatting of newlines with imports

- fixed unintentional folding of post scriptum standalone comments into last statement
 if it was a simple statement (#18, #28)

- fixed missing space in numpy-style array indexing (#33)

- fixed spurious space after star-based unary expressions (#31)

18.3a1

- added `--check`

- only put trailing commas in function signatures and calls if it's safe to do so. If
 the file is Python 3.6+ it's always safe, otherwise only safe if there are no `*args`
 or `**kwargs` used in the signature or call. (#8)

- fixed invalid spacing of dots in relative imports (#6, #13)

- fixed invalid splitting after comma on unpacked variables in for-loops (#23)

- fixed spurious space in parenthesized set expressions (#7)

- fixed spurious space after opening parentheses and in default arguments (#14, #17)

- fixed spurious space after unary operators when the operand was a complex expression
 (#15)

18.3a0

- first published version, Happy Day 2018!

- alpha quality

- date-versioned (see: <https://calver.org/>)
name: Lint

on: [push, pull_request]

jobs:
 build:
   # We want to run on external PRs, but not on our own internal PRs as they'll be run
   # by the push to the branch. Without this if check, checks are duplicated since
   # internal PRs match both the push and pull_request events.
   if:
     github.event_name == 'push' || github.event.pull_request.head.repo.full_name !=
     github.repository

   runs-on: ubuntu-latest

   steps:
     - uses: actions/checkout@v3

     - name: Assert PR target is main
       if: github.event_name == 'pull_request' && github.repository == 'psf/black'
       run: |
         if ["$GITHUB_BASE_REF" != "main"]; then
             echo "::error::PR targeting '$GITHUB_BASE_REF', please refile targeting 'main'." && exit 1
         fi

     - name: Set up latest Python
       uses: actions/setup-python@v4
       with:
         python-version: "*"

     - name: Install dependencies
       run: |
         python -m pip install --upgrade pip
         python -m pip install -e '.[d]'
         python -m pip install tox

     - name: Run pre-commit hooks
       uses: pre-commit/action@v3.0.0

     - name: Format ourselves
       run: |
         tox -e run_self

[![Black Logo](https://raw.githubusercontent.com/psf/black/main/docs/_static/logo2-readme.png)](https://black.readthedocs.io/en/stable/)

<h2 align="center">The Uncompromising Code Formatter</h2>

<p align="center">
<img alt="Actions Status"
src="https://github.com/psf/black/workflows/Test/badge.svg">
<img alt="Documentation Status"
src="https://readthedocs.org/projects/black/badge/?version=stable">
<img alt="Coverage Status"
src="https://coveralls.io/repos/github/psf/black/badge.svg?branch=main">
<img alt="License: MIT"
src="https://black.readthedocs.io/en/stable/_static/license.svg">
<img alt="PyPI" src="https://img.shields.io/pypi/v/black">
<img alt="Downloads" src="https://pepy.tech/badge/black">
<img alt="conda-forge"
src="https://img.shields.io/conda/dn/conda-forge/black.svg?label=conda-forge">
<img alt="Code style: black"
src="https://img.shields.io/badge/code%20style-black-000000.svg">
</p>

> Any color you like.

Black is the uncompromising Python code formatter. By using it, you agree to cede
control over minutiae of hand-formatting. In return, _Black_ gives you speed,
determinism, and freedom from `pycodestyle` nagging about formatting. You will save time
and mental energy for more important matters.

Blackened code looks the same regardless of the project you're reading. Formatting
becomes transparent after a while and you can focus on the content instead.

Black makes code review faster by producing the smallest diffs possible.

Try it out now using the [Black Playground](https://black.vercel.app). Watch the
[PyCon 2019 talk](https://youtu.be/esZLCuWs_2Y) to learn more.

[Read the documentation on ReadTheDocs!](https://black.readthedocs.io/en/stable)

Installation and usage

Installation

Black can be installed by running `pip install black`. It requires Python 3.7+ to run.
If you want to format Jupyter Notebooks, install with `pip install "black[jupyter]"`.

If you can't wait for the latest _hotness_ and want to install from GitHub, use:

`pip install git+https://github.com/psf/black`

Usage

To get started right away with sensible defaults:

```sh
black {source_file_or_directory}
```

You can run _Black_ as a package if running it as a script doesn't work:

```sh
python -m black {source_file_or_directory}
```

Further information can be found in our docs:

- [Usage and Configuration](https://black.readthedocs.io/en/stable/usage_and_configuration/index.html)

Black is already [successfully used](https://github.com/psf/black#used-by) by many
projects, small and big. _Black_ has a comprehensive test suite, with efficient parallel
tests, and our own auto formatting and parallel Continuous Integration runner. Now that
we have become stable, you should not expect large formatting changes in the future.
Stylistic changes will mostly be responses to bug reports and support for new Python
syntax. For more information please refer to the
[The Black Code Style](https://black.readthedocs.io/en/stable/the_black_code_style/index.html).

Also, as a safety measure which slows down processing, _Black_ will check that the
reformatted code still produces a valid AST that is effectively equivalent to the
original (see the
[Pragmatism](https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html#ast-before-and-after-
formatting)
section for details). If you're feeling confident, use `--fast`.

The _Black_ code style

Black is a PEP 8 compliant opinionated formatter. _Black_ reformats entire files in
place. Style configuration options are deliberately limited and rarely added. It doesn't
take previous formatting into account (see
[Pragmatism](https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html#pragmatism)
for exceptions).

Our documentation covers the current _Black_ code style, but planned changes to it are
also documented. They're both worth taking a look:

- [The _Black_ Code Style: Current
style](https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html)
- [The _Black_ Code Style: Future
style](https://black.readthedocs.io/en/stable/the_black_code_style/future_style.html)

Changes to the _Black_ code style are bound by the Stability Policy:

- [The _Black_ Code Style: Stability
Policy](https://black.readthedocs.io/en/stable/the_black_code_style/index.html#stability-policy)

Please refer to this document before submitting an issue. What seems like a bug might be
intended behaviour.

Pragmatism

Early versions of _Black_ used to be absolutist in some respects. They took after its
initial author. This was fine at the time as it made the implementation simpler and
there were not many users anyway. Not many edge cases were reported. As a mature tool,
Black does make some exceptions to rules it otherwise holds.

- [The _Black_ code style:
Pragmatism](https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html#pragmatism)

Please refer to this document before submitting an issue just like with the document
above. What seems like a bug might be intended behaviour.

Configuration

Black is able to read project-specific default values for its command line options
from a `pyproject.toml` file. This is especially useful for specifying custom
`--include` and `--exclude`/`--force-exclude`/`--extend-exclude` patterns for your
project.

You can find more details in our documentation:

- [The basics: Configuration via a
file](https://black.readthedocs.io/en/stable/usage_and_configuration/the_basics.html#configuration-via-a-file)

And if you're looking for more general configuration documentation:

- [Usage and Configuration](https://black.readthedocs.io/en/stable/usage_and_configuration/index.html)

Pro-tip: If you're asking yourself "Do I need to configure anything?" the answer is
"No". _Black_ is all about sensible defaults. Applying those defaults will have your
code in compliance with many other _Black_ formatted projects.

Used by

The following notable open-source projects trust _Black_ with enforcing a consistent
code style: pytest, tox, Pyramid, Django, Django Channels, Hypothesis, attrs,
SQLAlchemy, Poetry, PyPA applications (Warehouse, Bandersnatch, Pipenv, virtualenv),
pandas, Pillow, Twisted, LocalStack, every Datadog Agent Integration, Home Assistant,
Zulip, Kedro, OpenOA, FLORIS, ORBIT, WOMBAT, and many more.

The following organizations use _Black_: Facebook, Dropbox, KeepTruckin, Mozilla, Quora,
Duolingo, QuantumBlack, Tesla, Archer Aviation.

Are we missing anyone? Let us know.

Testimonials

Mike Bayer, [author of `SQLAlchemy`](https://www.sqlalchemy.org/):

> I can't think of any single tool in my entire programming career that has given me a
> bigger productivity increase by its introduction. I can now do refactorings in about
> 1% of the keystrokes that it would have taken me previously when we had no way for
> code to format itself.

Dusty Phillips,
[writer](https://smile.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=dusty+phillips):

> _Black_ is opinionated so you don't have to be.

Hynek Schlawack, [creator of `attrs`](https://www.attrs.org/), core developer of
Twisted and CPython:

> An auto-formatter that doesn't suck is all I want for Xmas!

Carl Meyer, [Django](https://www.djangoproject.com/) core developer:

> At least the name is good.

Kenneth Reitz, creator of [`requests`](https://requests.readthedocs.io/en/latest/)
and [`pipenv`](https://readthedocs.org/projects/pipenv/):

> This vastly improves the formatting of our code. Thanks a ton!

Show your style

Use the badge in your project's README.md:

```md
[![Code style: black](https://img.shields.io/badge/code%20style-black-000000.svg)](https://github.com/psf/black)
```

Using the badge in README.rst:

```
.. image:: https://img.shields.io/badge/code%20style-black-000000.svg
   :target: https://github.com/psf/black
```
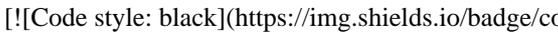
Looks like this:
[![Code style: black](https://img.shields.io/badge/code%20style-black-000000.svg)](https://github.com/psf/black)

License

MIT

Contributing

Welcome! Happy to see you willing to make the project better. You can get started by
reading this:

- [Contributing: The basics](https://black.readthedocs.io/en/latest/contributing/the_basics.html)

You can also take a look at the rest of the contributing docs or talk with the
developers:

- [Contributing documentation](https://black.readthedocs.io/en/latest/contributing/index.html)
- [Chat on Discord](https://discord.gg/RtVdv86PrH)

Change log

The log has become rather long. It moved to its own file.

See [CHANGES](https://black.readthedocs.io/en/latest/change_log.html).

Authors

The author list is quite long nowadays, so it lives in its own file.

See [AUTHORS.md](./AUTHORS.md)

Code of Conduct

Everyone participating in the _Black_ project, and in particular in the issue tracker,
pull requests, and social media activity, is expected to treat other people with respect
and more generally to follow the guidelines articulated in the
[Python Community Code of Conduct](https://www.python.org/psf/codeofconduct/).

At the same time, humor is encouraged. In fact, basic familiarity with Monty Python's
Flying Circus is expected. We are not savages.

And if you _really_ need to slap somebody, do it with a fish while dancing.
Black as a server (blackd)

`blackd` is a small HTTP server that exposes _Black_'s functionality over a simple
protocol. The main benefit of using it is to avoid the cost of starting up a new _Black_
process every time you want to blacken a file.

```{warning}
`blackd` should not be run as a publicly accessible server as there are no security
precautions in place to prevent abuse. **It is intended for local use only**.
```

Usage

`blackd` is not packaged alongside _Black_ by default because it has additional
dependencies. You will need to execute `pip install 'black[d]'` to install it.

You can start the server on the default port, binding only to the local interface by
running `blackd`. You will see a single line mentioning the server's version, and the
host and port it's listening on. `blackd` will then print an access log similar to most
web servers on standard output, merged with any exception traces caused by invalid
formatting requests.

`blackd` provides even less options than _Black_. You can see them by running
`blackd --help`:

```{program-output} blackd --help

```

There is no official `blackd` client tool (yet!). You can test that blackd is working
using `curl`:

```sh
blackd --bind-port 9090 &  # or let blackd choose a port
curl -s -XPOST "localhost:9090" -d "print('valid')"
```

Protocol

`blackd` only accepts `POST` requests at the `/` path. The body of the request should
contain the python source code to be formatted, encoded according to the `charset` field
in the `Content-Type` request header. If no `charset` is specified, `blackd` assumes
`UTF-8`.

There are a few HTTP headers that control how the source code is formatted. These
correspond to command line flags for _Black_. There is one exception to this:
`X-Protocol-Version` which if present, should have the value `1`, otherwise the request

is rejected with `HTTP 501` (Not Implemented).

The headers controlling how source code is formatted are:

- `X-Line-Length`: corresponds to the `--line-length` command line flag.
- `X-Skip-Source-First-Line`: corresponds to the `--skip-source-first-line` command line
  flag. If present and its value is not an empty string, the first line of the source
  code will be ignored.
- `X-Skip-String-Normalization`: corresponds to the `--skip-string-normalization`
  command line flag. If present and its value is not the empty string, no string
  normalization will be performed.
- `X-Skip-Magic-Trailing-Comma`: corresponds to the `--skip-magic-trailing-comma`
  command line flag. If present and its value is not an empty string, trailing commas
  will not be used as a reason to split lines.
- `X-Preview`: corresponds to the `--preview` command line flag. If present and its
  value is not an empty string, experimental and potentially disruptive style changes
  will be used.
- `X-Fast-Or-Safe`: if set to `fast`, `blackd` will act as _Black_ does when passed the
  `--fast` command line flag.
- `X-Python-Variant`: if set to `pyi`, `blackd` will act as _Black_ does when passed the
  `--pyi` command line flag. Otherwise, its value must correspond to a Python version or
  a set of comma-separated Python versions, optionally prefixed with `py`. For example,
  to request code that is compatible with Python 3.5 and 3.6, set the header to
  `py3.5,py3.6`.
- `X-Diff`: corresponds to the `--diff` command line flag. If present, a diff of the
  formats will be output.

If any of these headers are set to invalid values, `blackd` returns a `HTTP 400` error
response, mentioning the name of the problematic header in the message body.

Apart from the above, `blackd` can produce the following response codes:

- `HTTP 204`: If the input is already well-formatted. The response body is empty.
- `HTTP 200`: If formatting was needed on the input. The response body contains the
  blackened Python code, and the `Content-Type` header is set accordingly.
- `HTTP 400`: If the input contains a syntax error. Details of the error are returned in
  the response body.
- `HTTP 500`: If there was any other kind of error while trying to format the input. The
  response body contains a textual representation of the error.

The response headers include a `X-Black-Version` header containing the version of
Black.
[project]
name = "test"
version = "1.0.0"
requires-python = ">=3.7,<3.11"

[tool.black]

line-length = 79
Version control integration

Use [pre-commit](https://pre-commit.com/). Once you
[have it installed](https://pre-commit.com/#install), add this to the
`.pre-commit-config.yaml` in your repository:

```yaml
repos:
 - repo: https://github.com/psf/black
   rev: 23.1.0
   hooks:
     - id: black
       # It is recommended to specify the latest version of Python
       # supported by your project here, or alternatively use
       # pre-commit's default_language_version, see
       # https://pre-commit.com/#top_level-default_language_version
       language_version: python3.9
```

Feel free to switch out the `rev` value to something else, like another
[tag/version][black-tags] or even a specific commit. Although we discourage the use of
branches or other mutable refs since the hook [won't auto update as you may
expect][pre-commit-mutable-rev].

If you want support for Jupyter Notebooks as well, then replace `id: black` with
`id: black-jupyter`.

```{note}
The `black-jupyter` hook is only available from version 21.8b0 and onwards.
```

[black-tags]: https://github.com/psf/black/tags
[pre-commit-mutable-rev]:
 https://pre-commit.com/#using-the-latest-version-for-a-repository
FROM python:3.8.2-slim

note: a single RUN to avoid too many image layers being produced
RUN apt-get update \
&& apt-get upgrade -y \
&& apt-get install git apt-utils -y \
&& git config --global user.email "black@psf.github.com" \
&& git config --global user.name "Gallery/Black"

COPY gallery.py /
ENTRYPOINT ["python", "/gallery.py"]
name: docker

```
on:
 push:
  branches:
    - "main"
 release:
  types: [published]


permissions:
 contents: read


jobs:
 docker:
  if: github.repository == 'psf/black'
  runs-on: ubuntu-latest
  steps:
    - name: Checkout
      uses: actions/checkout@v3

    - name: Set up QEMU
      uses: docker/setup-qemu-action@v2

    - name: Set up Docker Buildx
      uses: docker/setup-buildx-action@v2

    - name: Login to DockerHub
      uses: docker/login-action@v2
      with:
       username: ${{ secrets.DOCKERHUB_USERNAME }}
       password: ${{ secrets.DOCKERHUB_TOKEN }}

   - name: Check + set version tag
     run:
       echo "GIT_TAG=$(git describe --candidates=0 --tags 2> /dev/null || echo
       latest_non_release)" >> $GITHUB_ENV

   - name: Build and push
     uses: docker/build-push-action@v3
     with:
       context: .
       platforms: linux/amd64,linux/arm64
       push: true
       tags: pyfound/black:latest,pyfound/black:${{ env.GIT_TAG }}

   - name: Build and push latest_release tag
     if:
       ${{ github.event_name == 'release' && github.event.action == 'published' &&
       !github.event.release.prerelease }}
     uses: docker/build-push-action@v3
```

```
        with:
          context: .
          platforms: linux/amd64,linux/arm64
          push: true
          tags: pyfound/black:latest_release

      - name: Build and push latest_prerelease tag
        if:
          ${{ github.event_name == 'release' && github.event.action == 'published' &&
          github.event.release.prerelease }}
        uses: docker/build-push-action@v3
        with:
          context: .
          platforms: linux/amd64,linux/arm64
          push: true
          tags: pyfound/black:latest_prerelease

      - name: Image digest
        run: echo ${{ steps.docker_build.outputs.digest }}
coverage >= 5.3
pre-commit
pytest >= 6.1.1
pytest-xdist >= 2.2.1, < 3.0.2
pytest-cov >= 2.11.1
tox
# Example configuration for Black.

# NOTE: you have to use single-quoted strings in TOML for regular expressions.
# It's the equivalent of r-strings in Python.  Multiline strings are treated as
# verbose regular expressions by Black.  Use [ ] to denote a significant space
# character.

[tool.black]
line-length = 88
target-version = ['py37', 'py38']
include = '\.pyi?$'
extend-exclude = '''
/(
  # The following are specific to Black, you probably don't want those.
  | blib2to3
  | tests/data
  | profiling
)/
'''
# We use preview style for formatting Black itself. If you
# want stable formatting across releases, you should keep
# this off.
preview = true
```

```
# Build system information and other project-specific configuration below.
# NOTE: You don't need this in your own Black configuration.

[build-system]
requires = ["hatchling>=1.8.0", "hatch-vcs", "hatch-fancy-pypi-readme"]
build-backend = "hatchling.build"

[project]
name = "black"
description = "The uncompromising code formatter."
license = { text = "MIT" }
requires-python = ">=3.7"
authors = [
  { name = "ukasz Langa", email = "lukasz@langa.pl" },
]
keywords = [
  "automation",
  "autopep8",
  "formatter",
  "gofmt",
  "pyfmt",
  "rustfmt",
  "yapf",
]
classifiers = [
  "Development Status :: 5 - Production/Stable",
  "Environment :: Console",
  "Intended Audience :: Developers",
  "License :: OSI Approved :: MIT License",
  "Operating System :: OS Independent",
  "Programming Language :: Python",
  "Programming Language :: Python :: 3 :: Only",
  "Programming Language :: Python :: 3.7",
  "Programming Language :: Python :: 3.8",
  "Programming Language :: Python :: 3.9",
  "Programming Language :: Python :: 3.10",
  "Programming Language :: Python :: 3.11",
  "Topic :: Software Development :: Libraries :: Python Modules",
  "Topic :: Software Development :: Quality Assurance",
]
dependencies = [
  "click>=8.0.0",
  "mypy_extensions>=0.4.3",
  "packaging>=22.0",
  "pathspec>=0.9.0",
  "platformdirs>=2",
  "tomli>=1.1.0; python_version < '3.11'",
```

```
  "typed-ast>=1.4.2; python_version < '3.8' and implementation_name == 'cpython'",
  "typing_extensions>=3.10.0.0; python_version < '3.10'",
]
dynamic = ["readme", "version"]

[project.optional-dependencies]
colorama = ["colorama>=0.4.3"]
uvloop = ["uvloop>=0.15.2"]
d = [
 "aiohttp>=3.7.4",
]
jupyter = [
 "ipython>=7.8.0",
 "tokenize-rt>=3.2.0",
]

[project.scripts]
black = "black:patched_main"
blackd = "blackd:patched_main [d]"

[project.urls]
Changelog = "https://github.com/psf/black/blob/main/CHANGES.md"
Homepage = "https://github.com/psf/black"

[tool.hatch.metadata.hooks.fancy-pypi-readme]
content-type = "text/markdown"
fragments = [
 { path = "README.md" },
 { path = "CHANGES.md" },
]

[tool.hatch.version]
source = "vcs"

[tool.hatch.build.hooks.vcs]
version-file = "src/_black_version.py"
template = '''
version = "{version}"
'''

[tool.hatch.build.targets.sdist]
exclude = ["/profiling"]

[tool.hatch.build.targets.wheel]
only-include = ["src"]
sources = ["src"]

[tool.hatch.build.targets.wheel.hooks.mypyc]
```

```
enable-by-default = false
dependencies = [
 "hatch-mypyc>=0.13.0",
 "mypy==0.991",
 # Required stubs to be removed when the packages support PEP 561 themselves
 "types-typed-ast>=1.4.2",
]
require-runtime-dependencies = true
exclude = [
 # There's no good reason for blackd to be compiled.
 "/src/blackd",
 # Not performance sensitive, so save bytes + compilation time:
 "/src/blib2to3/__init__.py",
 "/src/blib2to3/pgen2/__init__.py",
 "/src/black/output.py",
 "/src/black/concurrency.py",
 "/src/black/files.py",
 "/src/black/report.py",
 # Breaks the test suite when compiled (and is also useless):
 "/src/black/debug.py",
 # Compiled modules can't be run directly and that's a problem here:
 "/src/black/__main__.py",
]
options = { debug_level = "0" }


[tool.cibuildwheel]
build-verbosity = 1
# So these are the environments we target:
# - Python: CPython 3.7+ only
# - Architecture (64-bit only): amd64 / x86_64, universal2, and arm64
# - OS: Linux (no musl), Windows, and macOS
build = "cp3*-*"
skip = ["*-manylinux_i686", "*-musllinux_*", "*-win32", "pp-*"]
# This is the bare minimum needed to run the test suite. Pulling in the full
# test_requirements.txt would download a bunch of other packages not necessary
# here and would slow down the testing step a fair bit.
test-requires = ["pytest>=6.1.1"]
test-command = 'pytest {project} -k "not incompatible_with_mypyc"'
test-extras = ["d"," jupyter"]
# Skip trying to test arm64 builds on Intel Macs. (so cross-compilation doesn't
# straight up crash)
test-skip = ["*-macosx_arm64", "*-macosx_universal2:arm64"]


[tool.cibuildwheel.environment]
HATCH_BUILD_HOOKS_ENABLE = "1"
MYPYC_OPT_LEVEL = "3"
MYPYC_DEBUG_LEVEL = "0"
# CPython 3.11 wheels aren't available for aiohttp and building a Cython extension
```

```
# from source also doesn't work.
AIOHTTP_NO_EXTENSIONS = "1"

[tool.cibuildwheel.linux]
before-build = [
  "yum install -y clang gcc",
]

[tool.cibuildwheel.linux.environment]
HATCH_BUILD_HOOKS_ENABLE = "1"
MYPYC_OPT_LEVEL = "3"
MYPYC_DEBUG_LEVEL = "0"
# Black needs Clang to compile successfully on Linux.
CC = "clang"
AIOHTTP_NO_EXTENSIONS = "1"

[tool.isort]
atomic = true
profile = "black"
line_length = 88
skip_gitignore = true
skip_glob = ["src/blib2to3", "tests/data", "profiling"]
known_first_party = ["black", "blib2to3", "blackd", "_black_version"]

[tool.pytest.ini_options]
# Option below requires `tests/optional.py`
addopts = "--strict-config --strict-markers"
optional-tests = [
 "no_blackd: run when `d` extra NOT installed",
 "no_jupyter: run when `jupyter` extra NOT installed",
]
markers = [
 "incompatible_with_mypyc: run when testing mypyc compiled black"
]
xfail_strict = true
filterwarnings = [
  "error",
  # this is mitigated by a try/catch in https://github.com/psf/black/pull/2974/
  # this ignore can be removed when support for aiohttp 3.7 is dropped.
  "'ignore:Decorator `@unittest_run_loop` is no longer needed in aiohttp 3\.8\+:DeprecationWarning'",
  # this is mitigated by a try/catch in https://github.com/psf/black/pull/3198/
  # this ignore can be removed when support for aiohttp 3.x is dropped.
  "'ignore:Middleware decorator is deprecated since 4\.0 and its behaviour is default, you can simply remove this
decorator:DeprecationWarning'",
  # this is mitigated by https://github.com/python/cpython/issues/79071 in python 3.8+
  # this ignore can be removed when support for 3.7 is dropped.
  "'ignore:Bare functions are deprecated, use async ones:DeprecationWarning'",
  # aiohttp is using deprecated cgi modules - Safe to remove when fixed:
```

```
    # https://github.com/aio-libs/aiohttp/issues/6905
    "'ignore:'cgi' is deprecated and slated for removal in Python 3.13:DeprecationWarning'",
]
```
The basics

An overview on contributing to the _Black_ project.

Technicalities

Development on the latest version of Python is preferred. You can use any operating system.

Install development dependencies inside a virtual environment of your choice, for example:

```console
$ python3 -m venv .venv
$ source .venv/bin/activate
(.venv)$ pip install -r test_requirements.txt
(.venv)$ pip install -e .[d]
(.venv)$ pre-commit install
```

Before submitting pull requests, run lints and tests with the following commands from the root of the black repo:

```console
# Linting
(.venv)$ pre-commit run -a

# Unit tests
(.venv)$ tox -e py

# Optional Fuzz testing
(.venv)$ tox -e fuzz
```

News / Changelog Requirement

`Black` has CI that will check for an entry corresponding to your PR in `CHANGES.md`. If you feel this PR does not require a changelog entry please state that in a comment and a maintainer can add a `skip news` label to make the CI pass. Otherwise, please ensure you have a line in the following format:

```md
- `Black` is now more awesome (#X)
```

Note that X should be your PR number, not issue number! To workout X, please use
[Next PR Number](https://ichard26.github.io/next-pr-number/?owner=psf&name=black). This
is not perfect but saves a lot of release overhead as now the releaser does not need to
go back and workout what to add to the `CHANGES.md` for each release.

Style Changes

If a change would affect the advertised code style, please modify the documentation (The
Black code style) to reflect that change. Patches that fix unintended bugs in
formatting don't need to be mentioned separately though. If the change is implemented
with the `--preview` flag, please include the change in the future style document
instead and write the changelog entry under a dedicated "Preview changes" heading.

Docs Testing

If you make changes to docs, you can test they still build locally too.

```console
(.venv)$ pip install -r docs/requirements.txt
(.venv)$ pip install [-e] .[d]
(.venv)$ sphinx-build -a -b html -W docs/ docs/_build/
```

Hygiene

If you're fixing a bug, add a test. Run it first to confirm it fails, then fix the bug,
run it again to confirm it's really fixed.

If adding a new feature, add a test. In fact, always add a test. But wait, before adding
any large feature, first open an issue for us to discuss the idea first.

Finally

Thanks again for your interest in improving the project! You're taking action when most
people decide to sit and watch.

Gallery

Gallery is a script that automates the process of applying different _Black_ versions to
a selected PyPI package and seeing the results between _Black_ versions.

Build

```console
$ docker build -t black_gallery .
```

Run

````console
$ docker run -it -v /host/output:/output -v /host/input:/input black_gallery:latest [args]
````

```
usage: gallery.py [-h] (-p PYPI_PACKAGE | -t TOP_PACKAGES) [-b BLACK_REPO] [-v VERSION] [-w
WORKERS] [-i INPUT] [-o OUTPUT]
            [versions [versions ...]]

Black Gallery is a script that automates the process of applying different Black versions to a selected PyPI package
and
seeing the results between versions.

positional arguments:
 versions

optional arguments:
 -h, --help          show this help message and exit
 -p PYPI_PACKAGE, --pypi-package PYPI_PACKAGE
               PyPI package to download.
 -t TOP_PACKAGES, --top-packages TOP_PACKAGES
               Top n PyPI packages to download.
 -b BLACK_REPO, --black-repo BLACK_REPO
               Black's Git repository.
 -v VERSION, --version VERSION
               Version for given PyPI package. Will be discarded if used with -t option.
 -w WORKERS, --workers WORKERS
               Maximum number of threads to download with at the same time. Will be discarded if used with -p
               option.
 -i INPUT, --input INPUT
               Input directory to read configuration.
 -o OUTPUT, --output OUTPUT
               Output directory to download and put result artifacts.
```

{"metadata":{"kernelspec":{"name":"ir","display_name":"R","language":"R"},"language_info":{"name":"R","code
mirror_mode":"r","pygments_lexer":"r","mimetype":"text/x-r-
source","file_extension":".r","version":"4.0.5"}},"nbformat_minor":4,"nbformat":4,"cells":[{"cell_type":"code","so
urce":"library(tidyverse)
","metadata":{"_uuid":"051d70d956493feee0c6d64651c6a088724dca2a","_execution_state":"idle"},"execution_cou
nt":null,"outputs":[]}]}
Empty pyproject.toml to use with some tests that depend on Python 3.6 autodiscovery
and so on.
name: Test

on:
 push:
  paths-ignore:
   - "docs/**"

```yaml
      - "*.md"

  pull_request:
    paths-ignore:
      - "docs/**"
      - "*.md"

permissions:
  contents: read

concurrency:
  group: ${{ github.workflow }}-${{ github.event.pull_request.number || github.run_id }}
  cancel-in-progress: true

jobs:
  main:
    # We want to run on external PRs, but not on our own internal PRs as they'll be run
    # by the push to the branch. Without this if check, checks are duplicated since
    # internal PRs match both the push and pull_request events.
    if:
      github.event_name == 'push' || github.event.pull_request.head.repo.full_name !=
      github.repository

    runs-on: ${{ matrix.os }}
    strategy:
      fail-fast: false
      matrix:
        python-version: ["3.7", "3.8", "3.9", "3.10", "3.11", "pypy-3.7", "pypy-3.8"]
        os: [ubuntu-latest, macOS-latest, windows-latest]

    steps:
      - uses: actions/checkout@v3

      - name: Set up Python ${{ matrix.python-version }}
        uses: actions/setup-python@v4
        with:
          python-version: ${{ matrix.python-version }}

      - name: Install tox
        run: |
          python -m pip install --upgrade pip
          python -m pip install --upgrade tox

      - name: Unit tests
        if: "!startsWith(matrix.python-version, 'pypy')"
        run: tox -e ci-py -- -v --color=yes

      - name: Unit tests (pypy)
```

```
      if: "startsWith(matrix.python-version, 'pypy')"
      run: tox -e ci-pypy3 -- -v --color=yes


    - name: Upload coverage to Coveralls
      # Upload coverage if we are on the main repository and
      # we're running on Linux (this action only supports Linux)
      if: github.repository == 'psf/black' && matrix.os == 'ubuntu-latest'
      uses: AndreMiras/coveralls-python-action@v20201129
      with:
        github-token: ${{ secrets.GITHUB_TOKEN }}
        parallel: true
        flag-name: py${{ matrix.python-version }}-${{ matrix.os }}
        debug: true


  coveralls-finish:
    needs: main
    if: github.repository == 'psf/black'


    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - name: Send finished signal to Coveralls
        uses: AndreMiras/coveralls-python-action@v20201129
        with:
          parallel-finished: true
          debug: true


  uvloop:
    if:
      github.event_name == 'push' || github.event.pull_request.head.repo.full_name !=
      github.repository
    runs-on: ${{ matrix.os }}
    strategy:
      fail-fast: false
      matrix:
        os: [ubuntu-latest, macOS-latest]


    steps:
      - uses: actions/checkout@v3


      - name: Set up latest Python
        uses: actions/setup-python@v4
        with:
          python-version: "*"


      - name: Install black with uvloop
        run: |
          python -m pip install pip --upgrade --disable-pip-version-check
```

```
      python -m pip install -e ".[uvloop]"

  - name: Format ourselves
    run: python -m black --check src/
```
---
name: Feature request
about: Suggest an idea for this project
title: ""
labels: "T: enhancement"
assignees: ""

Is your feature request related to a problem? Please describe.

<!-- A clear and concise description of what the problem is.
e.g. I'm always frustrated when [...] -->

Describe the solution you'd like

<!-- A clear and concise description of what you want to
happen. -->

Describe alternatives you've considered

<!-- A clear and concise description of any
alternative solutions or features you've considered. -->

Additional context

<!-- Add any other context or screenshots about the feature request
here. -->
[project]
name = "test"
version = "1.0.0"
requires-python = ">=3.7,<3.11"

[tool.black]
line-length = 79
target-version = ["py310"]
Integrations

```{toctree}
---
hidden:
---

editors
github_actions
```

source_version_control
```

_Black_ can be integrated into many environments, providing a better and smoother
experience. Documentation for integrating _Black_ with a tool can be found for the
following areas:

- {doc}`Editor / IDE <./editors>`
- {doc}`GitHub Actions <./github_actions>`
- {doc}`Source version control <./source_version_control>`

Editors and tools not listed will require external contributions.

Patches welcome!

Any tool can pipe code through _Black_ using its stdio mode (just
[use `-` as the file name](https://www.tldp.org/LDP/abs/html/special-chars.html#DASHREF2)).
The formatted code will be returned on stdout (unless `--check` was passed). _Black_
will still emit messages on stderr but that shouldn't affect your use case.

This can be used for example with PyCharm's or IntelliJ's
[File Watchers](https://www.jetbrains.com/help/pycharm/file-watchers.html).
# Treat each other well

Everyone participating in the _Black_ project, and in particular in the issue tracker,
pull requests, and social media activity, is expected to treat other people with respect
and more generally to follow the guidelines articulated in the
[Python Community Code of Conduct](https://www.python.org/psf/codeofconduct/).

At the same time, humor is encouraged. In fact, basic familiarity with Monty Python's
Flying Circus is expected. We are not savages.

And if you _really_ need to slap somebody, do it with a fish while dancing.
# https://docs.github.com/en/code-security/supply-chain-security/keeping-your-dependencies-updated-
automatically/configuration-options-for-dependency-updates

version: 2
updates:
 - package-ecosystem: "github-actions"
   # Workflow files in .github/workflows will be checked
   directory: "/"
   schedule:
     interval: "weekly"
   labels: ["skip news", "C: dependencies"]

 - package-ecosystem: "pip"
   directory: "docs/"
   schedule:

```
    interval: "weekly"
   labels: ["skip news", "C: dependencies", "T: documentation"]
   reviewers: ["ichard26"]
# The (future of the) Black code style
```

```{warning}
Changes to this document often aren't tied and don't relate to releases of
_Black_. It's recommended that you read the latest version available.
```

Using backslashes for with statements

[Backslashes are bad and should be never be used](labels/why-no-backslashes) however
there is one exception: `with` statements using multiple context managers. Before Python
3.9 Python's grammar does not allow organizing parentheses around the series of context
managers.

We don't want formatting like:

```py3
with make_context_manager1() as cm1, make_context_manager2() as cm2, make_context_manager3() as cm3,
make_context_manager4() as cm4:
    ... # nothing to split on - line too long
```

So _Black_ will, when we implement this, format it like this:

```py3
with \
   make_context_manager1() as cm1, \
   make_context_manager2() as cm2, \
   make_context_manager3() as cm3, \
   make_context_manager4() as cm4 \
:
   ... # backslashes and an ugly stranded colon
```

Although when the target version is Python 3.9 or higher, _Black_ uses parentheses
instead in `--preview` mode (see below) since they're allowed in Python 3.9 and higher.

An alternative to consider if the backslashes in the above formatting are undesirable is
to use {external:py:obj}`contextlib.ExitStack` to combine context managers in the
following way:

```python
with contextlib.ExitStack() as exit_stack:
   cm1 = exit_stack.enter_context(make_context_manager1())
   cm2 = exit_stack.enter_context(make_context_manager2())
```

```
    cm3 = exit_stack.enter_context(make_context_manager3())
    cm4 = exit_stack.enter_context(make_context_manager4())
    ...
```

Preview style

Experimental, potentially disruptive style changes are gathered under the `--preview`
CLI flag. At the end of each year, these changes may be adopted into the default style,
as described in [The Black Code Style](./index.rst). Because the functionality is
experimental, feedback and issue reports are highly encouraged!

Improved string processing

Black will split long string literals and merge short ones. Parentheses are used where
appropriate. When split, parts of f-strings that don't need formatting are converted to
plain strings. User-made splits are respected when they do not exceed the line length
limit. Line continuation backslashes are converted into parenthesized strings.
Unnecessary parentheses are stripped. The stability and status of this feature is
tracked in [this issue](https://github.com/psf/black/issues/2188).

```
{
"cells": [
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
   "tags": []
  },
  "outputs": [],
  "source": [
   "%%time\n",
   "\n",
   "print('foo')"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {},
  "outputs": [],
  "source": []
 }
],
"metadata": {},
"nbformat": 4,
"nbformat_minor": 4
}
```

---
orphan: true

License

```{include} ../LICENSE

```

Contributing

```{toctree}
---
hidden:
---

the_basics
gauging_changes
issue_triage
release_process
reference/reference_summary
```

Welcome! Happy to see you willing to make the project better. Have you read the entire
[user documentation](https://black.readthedocs.io/en/latest/) yet?

```{rubric} Bird's eye view

```

In terms of inspiration, _Black_ is about as configurable as _gofmt_ (which is to say,
not very). This is deliberate. _Black_ aims to provide a consistent style and take away
opportunities for arguing about style.

Bug reports and fixes are always welcome! Please follow the
[issue template on GitHub](https://github.com/psf/black/issues/new) for best results.

Before you suggest a new feature or configuration knob, ask yourself why you want it. If
it enables better integration with some workflow, fixes an inconsistency, speeds things
up, and so on - go for it! On the other hand, if your answer is "because I don't like a
particular formatting" then you're not ready to embrace _Black_ yet. Such changes are
unlikely to get accepted. You can still try but prepare to be disappointed.

```{rubric} Contents

```

This section covers the following topics:

- {doc}`the_basics`
- {doc}`gauging_changes`
- {doc}`release_process`
- {doc}`reference/reference_summary`

For an overview on contributing to the _Black_, please checkout {doc}`the_basics`.

If you need a reference of the functions, classes, etc. available to you while
developing _Black_, there's the {doc}`reference/reference_summary` docs.
<!--
black documentation master file, created by
sphinx-quickstart on Fri Mar 23 10:53:30 2018.
-->

The uncompromising code formatter

> Any color you like.

By using _Black_, you agree to cede control over minutiae of hand-formatting. In return,
Black gives you speed, determinism, and freedom from `pycodestyle` nagging about
formatting. You will save time and mental energy for more important matters.

Black makes code review faster by producing the smallest diffs possible. Blackened
code looks the same regardless of the project you're reading. Formatting becomes
transparent after a while and you can focus on the content instead.

Try it out now using the [Black Playground](https://black.vercel.app).

```{admonition} Note - Black is now stable!
*Black* is [successfully used](https://github.com/psf/black#used-by) by
many projects, small and big. *Black* has a comprehensive test suite, with efficient
parallel tests, our own auto formatting and parallel Continuous Integration runner.
Now that we have become stable, you should not expect large formatting to changes in
the future. Stylistic changes will mostly be responses to bug reports and support for new Python
syntax.

Also, as a safety measure which slows down processing, *Black* will check that the
reformatted code still produces a valid AST that is effectively equivalent to the
original (see the
[Pragmatism](./the_black_code_style/current_style.md#pragmatism)
section for details). If you're feeling confident, use `--fast`.
```

```{note}
{doc}`Black is licensed under the MIT license <license>`.
```

Testimonials

Mike Bayer, author of [SQLAlchemy](https://www.sqlalchemy.org/):

> _I can't think of any single tool in my entire programming career that has given me a
> bigger productivity increase by its introduction. I can now do refactorings in about
> 1% of the keystrokes that it would have taken me previously when we had no way for
> code to format itself._

Dusty Phillips,
[writer](https://smile.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=dusty+phillips):

> _Black is opinionated so you don't have to be._

Hynek Schlawack, creator of [attrs](https://www.attrs.org/), core developer of
Twisted and CPython:

> _An auto-formatter that doesn't suck is all I want for Xmas!_

Carl Meyer, [Django](https://www.djangoproject.com/) core developer:

> _At least the name is good._

Kenneth Reitz, creator of [requests](http://python-requests.org/) and
[pipenv](https://docs.pipenv.org/):

> _This vastly improves the formatting of our code. Thanks a ton!_

Show your style

Use the badge in your project's README.md:

```md
[![Code style: black](https://img.shields.io/badge/code%20style-black-000000.svg)](https://github.com/psf/black)
```

Using the badge in README.rst:

```rst
.. image:: https://img.shields.io/badge/code%20style-black-000000.svg
  :target: https://github.com/psf/black
```

Looks like this:

```{image} https://img.shields.io/badge/code%20style-black-000000.svg
:target: https://github.com/psf/black
```

Contents

```{toctree}
---
maxdepth: 3
includehidden:
---

the_black_code_style/index
```

```{toctree}
---
maxdepth: 3
includehidden:
caption: User Guide
---

getting_started
usage_and_configuration/index
integrations/index
guides/index
faq
```

```{toctree}
---
maxdepth: 2
includehidden:
caption: Development
---

contributing/index
change_log
authors
```

```{toctree}
---
hidden:
caption: Project Links
---

GitHub <https://github.com/psf/black>
PyPI <https://pypi.org/project/black>
Chat <https://discord.gg/RtVdv86PrH>
```

Indices and tables

- {ref}`genindex`
- {ref}`search`
[*.py]
profile = black
# GitHub Actions integration

You can use _Black_ within a GitHub Actions workflow without setting your own Python
environment. Great for enforcing that your code matches the _Black_ code style.

Compatibility

This action is known to support all GitHub-hosted runner OSes. In addition, only
published versions of _Black_ are supported (i.e. whatever is available on PyPI).

Finally, this action installs _Black_ with the `colorama` extra so the `--color` flag
should work fine.

Usage

Create a file named `.github/workflows/black.yml` inside your repository with:

```yaml
name: Lint

on: [push, pull_request]

jobs:
 lint:
   runs-on: ubuntu-latest
   steps:
     - uses: actions/checkout@v2
     - uses: psf/black@stable
```

We recommend the use of the `@stable` tag, but per version tags also exist if you prefer
that. Note that the action's version you select is independent of the version of _Black_
the action will use.

The version of _Black_ the action will use can be configured via `version`. This can be
any
[valid version specifier](https://packaging.python.org/en/latest/glossary/#term-Version-Specifier)
or just the version number if you want an exact version. The action defaults to the
latest release available on PyPI. Only versions available from PyPI are supported, so no
commit SHAs or branch names.

If you want to include Jupyter Notebooks, _Black_ must be installed with the `jupyter` extra. Installing the extra and including Jupyter Notebook files can be configured via `jupyter` (default is `false`).

You can also configure the arguments passed to _Black_ via `options` (defaults to `` `--check --diff` ``) and `src` (default is `` `.` ``). Please note that the [`--check` flag](labels/exit-code) is required so that the workflow fails if _Black_ finds files that need to be formatted.

Here's an example configuration:

```yaml
- uses: psf/black@stable
 with:
   options: "--check --verbose"
   src: "./src"
   jupyter: true
   version: "21.5b1"
```

If you want to match versions covered by Black's [stability policy](labels/stability-policy), you can use the compatible release operator (`` `~=` ``):

```yaml
- uses: psf/black@stable
 with:
   options: "--check --verbose"
   src: "./src"
   version: "~= 22.0"
```
<!-- Hello! Thanks for submitting a PR. To help make things go a bit more
   smoothly we would appreciate that you go through this template. -->

Description

<!-- Good things to put here include: reasoning for the change (please link
   any relevant issues!), any noteworthy (or hacky) choices to be aware of,
   or what the problem resolved here looked like ... we won't mind a ranty
   story :) -->

Checklist - did you ...

<!-- If any of the following items aren't relevant for your contribution
   please still tick them so we know you've gone through the checklist.

   All user-facing changes should get an entry. Otherwise, signal to us
   this should get the magical label to silence the CHANGELOG entry check.

Tests are required for bugfixes and new features. Documentation changes
are necessary for formatting and most enhancement changes. -->

- [ ] Add an entry in `CHANGES.md` if necessary?
- [ ] Add / update tests if necessary?
- [ ] Add new / update outdated documentation?

<!-- Just as a reminder, everyone in all psf/black spaces including PRs
must follow the PSF Code of Conduct (link below).

Finally, once again thanks for your time and effort. If you have any
feedback in regards to your experience contributing here, please
let us know!

Helpful links:

PSF COC: https://www.python.org/psf/conduct/
Contributing docs: https://black.readthedocs.io/en/latest/contributing/index.html
Chat on Python Discord: https://discord.gg/RtVdv86PrH -->
Used by ReadTheDocs; pinned requirements for stability.

myst-parser==0.18.1
Sphinx==5.3.0
# Older versions break Sphinx even though they're declared to be supported.
docutils==0.19
sphinxcontrib-programoutput==0.17
sphinx_copybutton==0.5.1
furo==2022.12.7
# The _Black_ code style

Code style

Black aims for consistency, generality, readability and reducing git diffs. Similar
language constructs are formatted with similar rules. Style configuration options are
deliberately limited and rarely added. Previous formatting is taken into account as
little as possible, with rare exceptions like the magic trailing comma. The coding style
used by _Black_ can be viewed as a strict subset of PEP 8.

Black reformats entire files in place. It doesn't reformat lines that end with
`# fmt: skip` or blocks that start with `# fmt: off` and end with `# fmt: on`.
`# fmt: on/off` must be on the same level of indentation and in the same block, meaning
no unindents beyond the initial indentation level between them. It also recognizes
[YAPF](https://github.com/google/yapf)'s block comments to the same effect, as a
courtesy for straddling code.

The rest of this document describes the current formatting style. If you're interested
in trying out where the style is heading, see [future style](./future_style.md) and try
running `black --preview`.

How _Black_ wraps lines

Black ignores previous formatting and applies uniform horizontal and vertical whitespace to your code. The rules for horizontal whitespace can be summarized as: do whatever makes `pycodestyle` happy.

As for vertical whitespace, _Black_ tries to render one full expression or simple statement per line. If this fits the allotted line length, great.

```py3
# in:

j = [1,
    2,
    3
]

# out:

j = [1, 2, 3]
```

If not, _Black_ will look at the contents of the first outer matching brackets and put that in a separate indented line.

```py3
# in:

ImportantClass.important_method(exc, limit, lookup_lines, capture_locals, extra_argument)

# out:

ImportantClass.important_method(
    exc, limit, lookup_lines, capture_locals, extra_argument
)
```

If that still doesn't fit the bill, it will decompose the internal expression further using the same rule, indenting matching brackets every time. If the contents of the matching brackets pair are comma-separated (like an argument list, or a dict literal, and so on) then _Black_ will first try to keep them on the same line with the matching brackets. If that doesn't work, it will put all of them in separate lines.

```py3
# in:

def very_important_function(template: str, *variables, file: os.PathLike, engine: str, header: bool = True, debug:
```

```
    bool = False):
        """Applies `variables` to the `template` and writes to `file`."""
        with open(file, 'w') as f:
            ...

# out:

def very_important_function(
    template: str,
    *variables,
    file: os.PathLike,
    engine: str,
    header: bool = True,
    debug: bool = False,
):
    """Applies `variables` to the `template` and writes to `file`."""
    with open(file, "w") as f:
        ...
```

If a data structure literal (tuple, list, set, dict) or a line of "from" imports cannot
fit in the allotted length, it's always split into one element per line. This minimizes
diffs as well as enables readers of code to find which commit introduced a particular
entry. This also makes _Black_ compatible with
[isort](../guides/using_black_with_other_tools.md#isort) with the ready-made `black`
profile or manual configuration.

You might have noticed that closing brackets are always dedented and that a trailing
comma is always added. Such formatting produces smaller diffs; when you add or remove an
element, it's always just one line. Also, having the closing bracket dedented provides a
clear delimiter between two distinct sections of the code that otherwise share the same
indentation level (like the arguments list and the docstring in the example above).

(labels/why-no-backslashes)=

Black prefers parentheses over backslashes, and will remove backslashes if found.

```py3
# in:

if some_short_rule1 \
 and some_short_rule2:
    ...

# out:

if some_short_rule1 and some_short_rule2:
 ...
```

```
# in:

if some_long_rule1 \
 and some_long_rule2:
   ...

# out:

if (
   some_long_rule1
   and some_long_rule2
):
   ...
```

Backslashes and multiline strings are one of the two places in the Python grammar that break significant indentation. You never need backslashes, they are used to force the grammar to accept breaks that would otherwise be parse errors. That makes them confusing to look at and brittle to modify. This is why _Black_ always gets rid of them.

If you're reaching for backslashes, that's a clear signal that you can do better if you slightly refactor your code. I hope some of the examples above show you that there are many ways in which you can do it.

Line length

You probably noticed the peculiar default line length. _Black_ defaults to 88 characters per line, which happens to be 10% over 80. This number was found to produce significantly shorter files than sticking with 80 (the most popular), or even 79 (used by the standard library). In general,
[90-ish seems like the wise choice](https://youtu.be/wf-BqAjZb8M?t=260).

If you're paid by the line of code you write, you can pass `--line-length` with a lower number. _Black_ will try to respect that. However, sometimes it won't be able to without breaking other rules. In those rare cases, auto-formatted code will exceed your allotted limit.

You can also increase it, but remember that people with sight disabilities find it harder to work with line lengths exceeding 100 characters. It also adversely affects side-by-side diff review on typical screen resolutions. Long lines also make it harder to present code neatly in documentation or talk slides.

If you're using Flake8, you can bump `max-line-length` to 88 and mostly forget about it. However, it's better if you use [Bugbear](https://github.com/PyCQA/flake8-bugbear)'s B950 warning instead of E501, and bump the max line length to 88 (or the `--line-length`

you used for black), which will align more with black's _"try to respect
`--line-length`, but don't become crazy if you can't"_. You'd do it like this:

```ini
[flake8]
max-line-length = 88
...
select = C,E,F,W,B,B950
extend-ignore = E203, E501
```

Explanation of why E203 is disabled can be found further in this documentation. And if
you're curious about the reasoning behind B950,
[Bugbear's documentation](https://github.com/PyCQA/flake8-bugbear#opinionated-warnings)
explains it. The tl;dr is "it's like highway speed limits, we won't bother you if you
overdo it by a few km/h".

If you're looking for a minimal, black-compatible flake8 configuration:

```ini
[flake8]
max-line-length = 88
extend-ignore = E203
```

Empty lines

Black avoids spurious vertical whitespace. This is in the spirit of PEP 8 which says
that in-function vertical whitespace should only be used sparingly.

Black will allow single empty lines inside functions, and single and double empty
lines on module level left by the original editors, except when they're within
parenthesized expressions. Since such expressions are always reformatted to fit minimal
space, this whitespace is lost. The other exception is that it will remove any empty
lines immediately following a statement that introduces a new indentation level.

```python
# in:

def foo():

    print("All the newlines above me should be deleted!")


if condition:

    print("No newline above me!")
```

```
    print("There is a newline above me, and that's OK!")


class Point:

    x: int
    y: int

# out:

def foo():
    print("All the newlines above me should be deleted!")


if condition:
    print("No newline above me!")

    print("There is a newline above me, and that's OK!")


class Point:
    x: int
    y: int
```

It will also insert proper spacing before and after function definitions. It's one line
before and after inner functions and two lines before and after module-level functions
and classes. _Black_ will not put empty lines between function/class definitions and
standalone comments that immediately precede the given function/class.

Black will enforce single empty lines between a class-level docstring and the first
following field or method. This conforms to
[PEP 257](https://www.python.org/dev/peps/pep-0257/#multi-line-docstrings).

Black won't insert empty lines after function docstrings unless that empty line is
required due to an inner function starting immediately after.

Comments

Black does not format comment contents, but it enforces two spaces between code and a
comment on the same line, and a space before the comment text begins. Some types of
comments that require specific spacing rules are respected: doc comments (`#: comment`),
section comments with long runs of hashes, and Spyder cells. Non-breaking spaces after
hashes are also preserved. Comments may sometimes be moved because of formatting
changes, which can break tools that assign special meaning to them. See
[AST before and after formatting](#ast-before-and-after-formatting) for more discussion.

Trailing commas

Black will add trailing commas to expressions that are split by comma where each element is on its own line. This includes function signatures.

One exception to adding trailing commas is function signatures containing `*`, `*args`, or `**kwargs`. In this case a trailing comma is only safe to use on Python 3.6. _Black_ will detect if your file is already 3.6+ only and use trailing commas in this situation. If you wonder how it knows, it looks for f-strings and existing use of trailing commas in function signatures that have stars in them. In other words, if you'd like a trailing comma in this situation and _Black_ didn't recognize it was safe to do so, put it there manually and _Black_ will keep it.

A pre-existing trailing comma informs _Black_ to always explode contents of the current bracket pair into one item per line. Read more about this in the [Pragmatism](#pragmatism) section below.

Strings

Black prefers double quotes (`"` and `"""`) over single quotes (`'` and `''`). It will replace the latter with the former as long as it does not result in more backslash escapes than before.

Black also standardizes string prefixes. Prefix characters are made lowercase with the exception of [capital "R" prefixes](#rstrings-and-rstrings), unicode literal markers (`u`) are removed because they are meaningless in Python 3, and in the case of multiple characters "r" is put first as in spoken language: "raw f-string".

The main reason to standardize on a single form of quotes is aesthetics. Having one kind of quotes everywhere reduces reader distraction. It will also enable a future version of _Black_ to merge consecutive string literals that ended up on the same line (see [#26](https://github.com/psf/black/issues/26) for details).

Why settle on double quotes? They anticipate apostrophes in English text. They match the docstring standard described in [PEP 257](https://www.python.org/dev/peps/pep-0257/#what-is-a-docstring). An empty string in double quotes (`""`) is impossible to confuse with a one double-quote regardless of fonts and syntax highlighting used. On top of this, double quotes for strings are consistent with C which Python interacts a lot with.

On certain keyboard layouts like US English, typing single quotes is a bit easier than double quotes. The latter requires use of the Shift key. My recommendation here is to keep using whatever is faster to type and let _Black_ handle the transformation.

If you are adopting _Black_ in a large project with pre-existing string conventions (like the popular ["single quotes for data, double quotes for human-readable strings"](https://stackoverflow.com/a/56190)), you can pass `--skip-string-normalization` on the command line. This is meant as an adoption helper, avoid using this for new projects.

Black also processes docstrings. Firstly the indentation of docstrings is corrected for both quotations and the text within, although relative indentation in the text is preserved. Superfluous trailing whitespace on each line and unnecessary new lines at the end of the docstring are removed. All leading tabs are converted to spaces, but tabs inside text are preserved. Whitespace leading and trailing one-line docstrings is removed.

Numeric literals

Black standardizes most numeric literals to use lowercase letters for the syntactic parts and uppercase letters for the digits themselves: `0xAB` instead of `0XAB` and `1e10` instead of `1E10`.

Line breaks & binary operators

Black will break a line before a binary operator when splitting a block of code over multiple lines. This is so that _Black_ is compliant with the recent changes in the [PEP 8](https://www.python.org/dev/peps/pep-0008/#should-a-line-break-before-or-after-a-binary-operator) style guide, which emphasizes that this approach improves readability.

Almost all operators will be surrounded by single spaces, the only exceptions are unary operators (`+`, `-`, and `~`), and power operators when both operands are simple. For powers, an operand is considered simple if it's only a NAME, numeric CONSTANT, or attribute access (chained attribute access is allowed), with or without a preceding unary operator.

```python
# For example, these won't be surrounded by whitespace
a = x**y
b = config.base**5.2
c = config.base**runtime.config.exponent
d = 2**5
e = 2**~5

# ... but these will be surrounded by whitespace
f = 2 ** get_exponent()
g = get_x() ** get_y()
h = config['base'] ** 2
```

Slices

PEP 8
[recommends](https://www.python.org/dev/peps/pep-0008/#whitespace-in-expressions-and-statements)
to treat `:` in slices as a binary operator with the lowest priority, and to leave an
equal amount of space on either side, except if a parameter is omitted (e.g.
`ham[1 + 1 :]`). It recommends no spaces around `:` operators for "simple expressions"

(`ham[lower:upper]`), and extra space for "complex expressions"
(`ham[lower : upper + offset]`). _Black_ treats anything more than variable names as
"complex" (`ham[lower : upper + 1]`). It also states that for extended slices, both `:`
operators have to have the same amount of spacing, except if a parameter is omitted
(`ham[1 + 1 ::]`). _Black_ enforces these rules consistently.

This behaviour may raise `E203 whitespace before ':`` warnings in style guide
enforcement tools like Flake8. Since `E203` is not PEP 8 compliant, you should tell
Flake8 to ignore these warnings.

Parentheses

Some parentheses are optional in the Python grammar. Any expression can be wrapped in a
pair of parentheses to form an atom. There are a few interesting cases:

- `if (...):`
- `while (...):`
- `for (...) in (...):`
- `assert (...), (...)`
- `from X import (...)`
- assignments like:
  - `target = (...)`
  - `target: type = (...)`
  - `some, *un, packing = (...)`
  - `augmented += (...)`

In those cases, parentheses are removed when the entire statement fits in one line, or
if the inner expression doesn't have any delimiters to further split on. If there is
only a single delimiter and the expression starts or ends with a bracket, the
parentheses can also be successfully omitted since the existing bracket pair will
organize the expression neatly anyway. Otherwise, the parentheses are added.

Please note that _Black_ does not add or remove any additional nested parentheses that
you might want to have for clarity or further code organization. For example those
parentheses are not going to be removed:

```py3
return not (this or that)
decision = (maybe.this() and values > 0) or (maybe.that() and values < 0)
```

Call chains

Some popular APIs, like ORMs, use call chaining. This API style is known as a
[fluent interface](https://en.wikipedia.org/wiki/Fluent_interface). _Black_ formats
those by treating dots that follow a call or an indexing operation like a very low
priority delimiter. It's easier to show the behavior than to explain it. Look at the
example:

```py3
def example(session):
    result = (
        session.query(models.Customer.id)
        .filter(
            models.Customer.account_id == account_id,
            models.Customer.email == email_address,
        )
        .order_by(models.Customer.id.asc())
        .all()
    )
```

Typing stub files

PEP 484 describes the syntax for type hints in Python. One of the use cases for typing
is providing type annotations for modules which cannot contain them directly (they might
be written in C, or they might be third-party, or their implementation may be overly
dynamic, and so on).

To solve this,
[stub files with the `.pyi` file extension](https://www.python.org/dev/peps/pep-0484/#stub-files)
can be used to describe typing information for an external module. Those stub files omit
the implementation of classes and functions they describe, instead they only contain the
structure of the file (listing globals, functions, and classes with their members). The
recommended code style for those files is more terse than PEP 8:

- prefer `...` on the same line as the class/function signature;
- avoid vertical whitespace between consecutive module-level functions, names, or
 methods and fields within a single class;
- use a single blank line between top-level class definitions, or none if the classes
 are very small.

Black enforces the above rules. There are additional guidelines for formatting `.pyi`
file that are not enforced yet but might be in a future version of the formatter:

- prefer `...` over `pass`;
- avoid using string literals in type annotations, stub files support forward references
 natively (like Python 3.7 code with `from __future__ import annotations`);
- use variable annotations instead of type comments, even for stubs that target older
 versions of Python.

Line endings

Black will normalize line endings (`\n` or `\r\n`) based on the first line ending of
the file.

Pragmatism

Early versions of _Black_ used to be absolutist in some respects. They took after its initial author. This was fine at the time as it made the implementation simpler and there were not many users anyway. Not many edge cases were reported. As a mature tool, _Black_ does make some exceptions to rules it otherwise holds. This section documents what those exceptions are and why this is the case.

The magic trailing comma

Black in general does not take existing formatting into account.

However, there are cases where you put a short collection or function call in your code but you anticipate it will grow in the future.

For example:

```py3
TRANSLATIONS = {
    "en_us": "English (US)",
    "pl_pl": "polski",
}
```

Early versions of _Black_ used to ruthlessly collapse those into one line (it fits!). Now, you can communicate that you don't want that by putting a trailing comma in the collection yourself. When you do, _Black_ will know to always explode your collection into one item per line.

How do you make it stop? Just delete that trailing comma and _Black_ will collapse your collection into one line if it fits.

If you must, you can recover the behaviour of early versions of _Black_ with the option `--skip-magic-trailing-comma` / `-C`.

r"strings" and R"strings"

Black normalizes string quotes as well as string prefixes, making them lowercase. One exception to this rule is r-strings. It turns out that the very popular [MagicPython](https://github.com/MagicStack/MagicPython/) syntax highlighter, used by default by (among others) GitHub and Visual Studio Code, differentiates between r-strings and R-strings. The former are syntax highlighted as regular expressions while the latter are treated as true raw strings with no special semantics.

AST before and after formatting

When run with `--safe` (the default), _Black_ checks that the code before and after is semantically equivalent. This check is done by comparing the AST of the source with the

AST of the target. There are three limited cases in which the AST does differ:

1. _Black_ cleans up leading and trailing whitespace of docstrings, re-indenting them if
   needed. It's been one of the most popular user-reported features for the formatter to
   fix whitespace issues with docstrings. While the result is technically an AST
   difference, due to the various possibilities of forming docstrings, all real-world
   uses of docstrings that we're aware of sanitize indentation and leading/trailing
   whitespace anyway.

1. _Black_ manages optional parentheses for some statements. In the case of the `del`
   statement, presence of wrapping parentheses or lack of thereof changes the resulting
   AST but is semantically equivalent in the interpreter.

1. _Black_ might move comments around, which includes type comments. Those are part of
   the AST as of Python 3.8. While the tool implements a number of special cases for
   those comments, there is no guarantee they will remain where they were in the source.
   Note that this doesn't change runtime behavior of the source code.

To put things in perspective, the code equivalence check is a feature of _Black_ which
other formatters don't implement at all. It is of crucial importance to us to ensure
code behaves the way it did before it got reformatted. We treat this as a feature and
there are no plans to relax this in the future. The exceptions enumerated above stem
from either user feedback or implementation details of the tool. In each case we made
due diligence to ensure that the AST divergence is of no practical consequence.
See also: https://docs.github.com/en/communities/using-templates-to-encourage-useful-issues-and-pull-
requests/configuring-issue-templates-for-your-repository#configuring-the-template-chooser

This is the default and blank issues are useful so let's keep 'em.
blank_issues_enabled: true

contact_links:
 - name: Chat on Python Discord
   url: https://discord.gg/RtVdv86PrH
   about: |
     User support, questions, and other lightweight requests can be
     handled via the #black-formatter text channel we have on Python
     Discord.
version: 2

formats:
 - htmlzip

build:
 os: ubuntu-22.04
 tools:
   python: "3.8"

python:

```
install:
  - requirements: docs/requirements.txt

  - method: pip
    path: .
    extra_requirements:
      - d
[tool.isort]
profile = 'black'
name: "Black"
description: "The uncompromising Python code formatter."
author: "ukasz Langa and contributors to Black"
inputs:
 options:
   description:
     "Options passed to Black. Use `black --help` to see available options. Default:
     '--check --diff'"
   required: false
   default: "--check --diff"
 src:
   description: "Source to run Black. Default: '.'"
   required: false
   default: "."
 jupyter:
   description:
     "Set this option to true to include Jupyter Notebook files. Default: false"
   required: false
   default: false
 black_args:
   description: "[DEPRECATED] Black input arguments."
   required: false
   default: ""
   deprecationMessage:
     "Input `with.black_args` is deprecated. Use `with.options` and `with.src` instead."
 version:
   description: 'Python Version specifier (PEP440) - e.g. "21.5b1"'
   required: false
   default: ""
branding:
 color: "black"
 icon: "check-circle"
runs:
 using: composite
 steps:
   - run: |
       if [ "$RUNNER_OS" == "Windows" ]; then
         python $GITHUB_ACTION_PATH/action/main.py
       else
```

```
        python3 $GITHUB_ACTION_PATH/action/main.py
      fi
    env:
      # TODO: Remove once https://github.com/actions/runner/issues/665 is fixed.
      INPUT_OPTIONS: ${{ inputs.options }}
      INPUT_SRC: ${{ inputs.src }}
      INPUT_JUPYTER: ${{ inputs.jupyter }}
      INPUT_BLACK_ARGS: ${{ inputs.black_args }}
      INPUT_VERSION: ${{ inputs.version }}
      pythonioencoding: utf-8
    shell: bash
```

The Black Code Style

````
```{toctree}
---
hidden:
---

Current style <current_style>
Future style <future_style>
```
````

Black is a PEP 8 compliant opinionated formatter with its own style.

While keeping the style unchanged throughout releases has always been a goal, the _Black_ code style isn't set in stone. It evolves to accommodate for new features in the Python language and, occasionally, in response to user feedback. Large-scale style preferences presented in {doc}`current_style` are very unlikely to change, but minor style aspects and details might change according to the stability policy presented below. Ongoing style considerations are tracked on GitHub with the [design](https://github.com/psf/black/labels/T%3A%20design) issue label.

(labels/stability-policy)=

Stability Policy

The following policy applies for the _Black_ code style, in non pre-release versions of _Black_:

- If code has been formatted with _Black_, it will remain unchanged when formatted with the same options using any other release in the same calendar year.

This means projects can safely use `black ~= 22.0` without worrying about formatting changes disrupting their project in 2022. We may still fix bugs where _Black_ crashes on some code, and make other improvements that do not affect formatting.

In rare cases, we may make changes affecting code that has not been previously formatted with _Black_. For example, we have had bugs where we accidentally removed

some comments. Such bugs can be fixed without breaking the stability policy.

- The first release in a new calendar year _may_ contain formatting changes, although
  these will be minimised as much as possible. This is to allow for improved formatting
  enabled by newer Python language syntax as well as due to improvements in the
  formatting logic.

- The `--preview` flag is exempt from this policy. There are no guarantees around the
  stability of the output with that flag passed into _Black_. This flag is intended for
  allowing experimentation with the proposed changes to the _Black_ code style.

Documentation for both the current and future styles can be found:

- {doc}`current_style`
- {doc}`future_style`
[flake8]
# B905 should be enabled when we drop support for 3.9
ignore = E203, E266, E501, W503, B905, B907
# line length is intentionally set to 80 here because black uses Bugbear
# See https://black.readthedocs.io/en/stable/the_black_code_style/current_style.html#line-length for more details
max-line-length = 80
max-complexity = 18
select = B,C,E,F,W,T4,B9
# Grammar for 2to3. This grammar supports Python 2.x and 3.x.

NOTE WELL: You should also follow all the steps listed at
https://devguide.python.org/grammar/

Start symbols for the grammar:
file_input is a module or sequence of commands read from an input file;
single_input is a single interactive statement;
eval_input is the input for the eval() and input() functions.
NB: compound_stmt in single_input is followed by extra NEWLINE!
file_input: (NEWLINE | stmt)* ENDMARKER
single_input: NEWLINE | simple_stmt | compound_stmt NEWLINE
eval_input: testlist NEWLINE* ENDMARKER

decorator: '@' namedexpr_test NEWLINE
decorators: decorator+
decorated: decorators (classdef | funcdef | async_funcdef)
async_funcdef: ASYNC funcdef
funcdef: 'def' NAME parameters ['->' test] ':' suite
parameters: '(' [typedargslist] ')'

The following definition for typedarglist is equivalent to this set of rules:
#
arguments = argument (',' argument)*
argument = tfpdef ['=' test]

```
#     kwargs = '**' tname [',']
#     args = '*' [tname_star]
#     kwonly_kwargs = (',' argument)* [',' [kwargs]]
#     args_kwonly_kwargs = args kwonly_kwargs | kwargs
#     poskeyword_args_kwonly_kwargs = arguments [',' [args_kwonly_kwargs]]
#     typedargslist_no_posonly  = poskeyword_args_kwonly_kwargs | args_kwonly_kwargs
#     typedarglist = arguments ',' '/' [',' [typedargslist_no_posonly]])|(typedargslist_no_posonly)"
#
# It needs to be fully expanded to allow our LL(1) parser to work on it.

typedargslist: tfpdef ['=' test] (',' tfpdef ['=' test])* ',' '/' [
                ',' [((tfpdef ['=' test] ',')* ('*' [tname_star] (',' tname ['=' test])*
                    [',' ['**' tname [',']]] | '**' tname [','])
                | tfpdef ['=' test] (',' tfpdef ['=' test])* [','])]
            ] | ((tfpdef ['=' test] ',')* ('*' [tname_star] (',' tname ['=' test])*
                [',' ['**' tname [',']]] | '**' tname [','])
                | tfpdef ['=' test] (',' tfpdef ['=' test])* [','])


tname: NAME [':' test]
tname_star: NAME [':' (test|star_expr)]
tfpdef: tname | '(' tfplist ')'
tfplist: tfpdef (',' tfpdef)* [',']


# The following definition for varargslist is equivalent to this set of rules:
#
#     arguments = argument (',' argument )*
#     argument = vfpdef ['=' test]
#     kwargs = '**' vname [',']
#     args = '*' [vname]
#     kwonly_kwargs = (',' argument )* [',' [kwargs]]
#     args_kwonly_kwargs = args kwonly_kwargs | kwargs
#     poskeyword_args_kwonly_kwargs = arguments [',' [args_kwonly_kwargs]]
#     vararglist_no_posonly = poskeyword_args_kwonly_kwargs | args_kwonly_kwargs
#     varargslist = arguments ',' '/' [',' [(vararglist_no_posonly)]] | (vararglist_no_posonly)
#
# It needs to be fully expanded to allow our LL(1) parser to work on it.

varargslist: vfpdef ['=' test ](',' vfpdef ['=' test])* ',' '/' [',' [
                ((vfpdef ['=' test] ',')* ('*' [vname] (',' vname ['=' test])*
                    [',' ['**' vname [',']]] | '**' vname [','])
                    | vfpdef ['=' test] (',' vfpdef ['=' test])* [','])
                ]] | ((vfpdef ['=' test] ',')*
                ('*' [vname] (',' vname ['=' test])*  [',' ['**' vname [',']]]| '**' vname [','])
                | vfpdef ['=' test] (',' vfpdef ['=' test])* [','])


vname: NAME
vfpdef: vname | '(' vfplist ')'
vfplist: vfpdef (',' vfpdef)* [',']
```

stmt: simple_stmt | compound_stmt

simple_stmt: small_stmt (';' small_stmt)* [';'] NEWLINE

small_stmt: (expr_stmt | print_stmt | del_stmt | pass_stmt | flow_stmt |

        import_stmt | global_stmt | exec_stmt | assert_stmt)

expr_stmt: testlist_star_expr (annassign | augassign (yield_expr|testlist) |

          ('=' (yield_expr|testlist_star_expr))*)

annassign: ':' test ['=' (yield_expr|testlist_star_expr)]

testlist_star_expr: (test|star_expr) (',' (test|star_expr))* [',']

augassign: ('+=' | '-=' | '*=' | '@=' | '/=' | '%=' | '&=' | '|=' | '^=' |

      '<<=' | '>>=' | '**=' | '//=')

For normal and annotated assignments, additional restrictions enforced by the interpreter

print_stmt: 'print' ([test (',' test)* [',']] |

             '>>' test [(',' test)+ [',']])

del_stmt: 'del' exprlist

pass_stmt: 'pass'

flow_stmt: break_stmt | continue_stmt | return_stmt | raise_stmt | yield_stmt

break_stmt: 'break'

continue_stmt: 'continue'

return_stmt: 'return' [testlist_star_expr]

yield_stmt: yield_expr

raise_stmt: 'raise' [test ['from' test | ',' test [',' test]]]

import_stmt: import_name | import_from

import_name: 'import' dotted_as_names

import_from: ('from' ('.'* dotted_name | '.'+)

      'import' ('*' | '(' import_as_names ')' | import_as_names))

import_as_name: NAME ['as' NAME]

dotted_as_name: dotted_name ['as' NAME]

import_as_names: import_as_name (',' import_as_name)* [',']

dotted_as_names: dotted_as_name (',' dotted_as_name)*

dotted_name: NAME ('.' NAME)*

global_stmt: ('global' | 'nonlocal') NAME (',' NAME)*

exec_stmt: 'exec' expr ['in' test [',' test]]

assert_stmt: 'assert' test [',' test]

compound_stmt: if_stmt | while_stmt | for_stmt | try_stmt | with_stmt | funcdef | classdef | decorated | async_stmt |
match_stmt

async_stmt: ASYNC (funcdef | with_stmt | for_stmt)

if_stmt: 'if' namedexpr_test ':' suite ('elif' namedexpr_test ':' suite)* ['else' ':' suite]

while_stmt: 'while' namedexpr_test ':' suite ['else' ':' suite]

for_stmt: 'for' exprlist 'in' testlist_star_expr ':' suite ['else' ':' suite]

try_stmt: ('try' ':' suite

     ((except_clause ':' suite)+

   ['else' ':' suite]

   ['finally' ':' suite] |

   'finally' ':' suite))

with_stmt: 'with' asexpr_test (',' asexpr_test)* ':' suite

NB compile.c makes sure that the default except clause is last
except_clause: 'except' ['*'] [test [(',' | 'as') test]]
suite: simple_stmt | NEWLINE INDENT stmt+ DEDENT

Backward compatibility cruft to support:
[x for x in lambda: True, lambda: False if x()]
even while also allowing:
lambda x: 5 if x else 2
(But not a mix of the two)
testlist_safe: old_test [(',' old_test)+ [',']]
old_test: or_test | old_lambdef
old_lambdef: 'lambda' [varargslist] ':' old_test

namedexpr_test: asexpr_test [':=' asexpr_test]

This is actually not a real rule, though since the parser is very
limited in terms of the strategy about match/case rules, we are inserting
a virtual case (<expr> as <expr>) as a valid expression. Unless a better
approach is thought, the only side effect of this seem to be just allowing
more stuff to be parser (which would fail on the ast).
asexpr_test: test ['as' test]

test: or_test ['if' or_test 'else' test] | lambdef
or_test: and_test ('or' and_test)*
and_test: not_test ('and' not_test)*
not_test: 'not' not_test | comparison
comparison: expr (comp_op expr)*
comp_op: '<'|'>'|'=='|'>='|'<='|'<>'|'!='|'in'|'not' 'in'|'is'|'is' 'not'
star_expr: '*' expr
expr: xor_expr ('|' xor_expr)*
xor_expr: and_expr ('^' and_expr)*
and_expr: shift_expr ('&' shift_expr)*
shift_expr: arith_expr (('<<'|'>>') arith_expr)*
arith_expr: term (('+'|'-') term)*
term: factor (('*'|'@'|'/'|'%'|'//') factor)*
factor: ('+'|'-'|'~') factor | power
power: [AWAIT] atom trailer* ['**' factor]
atom: ('(' [yield_expr|testlist_gexp] ')' |
    '[' [listmaker] ']' |
    '{' [dictsetmaker] '}' |
    '`' testlist1 '`' |
    NAME | NUMBER | STRING+ | '.' '.' '.')
listmaker: (namedexpr_test|star_expr) (old_comp_for | (',' (namedexpr_test|star_expr))* [','])
testlist_gexp: (namedexpr_test|star_expr) (old_comp_for | (',' (namedexpr_test|star_expr))* [','])
lambdef: 'lambda' [varargslist] ':' test
trailer: '(' [arglist] ')' | '[' subscriptlist ']' | '.' NAME
subscriptlist: (subscript|star_expr) (',' (subscript|star_expr))* [',']
subscript: test [':=' test] | [test] ':' [test] [sliceop]

```
sliceop: ':' [test]
exprlist: (expr|star_expr) (',' (expr|star_expr))* [',']
testlist: test (',' test)* [',']
dictsetmaker: ( (((test ':' asexpr_test | '**' expr)
            (comp_for | (',' (test ':' asexpr_test | '**' expr))* [','])) |
          ((test [':=' test] | star_expr)
   (comp_for | (',' (test [':=' test] | star_expr))* [','])) )


classdef: 'class' NAME ['(' [arglist] ')'] ':' suite


arglist: argument (',' argument)* [',']


# "test '=' test" is really "keyword '=' test", but we have no such token.
# These need to be in a single rule to avoid grammar that is ambiguous
# to our LL(1) parser. Even though 'test' includes '*expr' in star_expr,
# we explicitly match '*' here, too, to give it proper precedence.
# Illegal combinations and orderings are blocked in ast.c:
# multiple (test comp_for) arguments are blocked; keyword unpackings
# that precede iterable unpackings are blocked; etc.
argument: ( test [comp_for] |
        test ':=' test [comp_for] |
        test 'as' test |
        test '=' asexpr_test |
   '**' test |
        '*' test )


comp_iter: comp_for | comp_if
comp_for: [ASYNC] 'for' exprlist 'in' or_test [comp_iter]
comp_if: 'if' old_test [comp_iter]


# As noted above, testlist_safe extends the syntax allowed in list
# comprehensions and generators. We can't use it indiscriminately in all
# derivations using a comp_for-like pattern because the testlist_safe derivation
# contains comma which clashes with trailing comma in arglist.
#
# This was an issue because the parser would not follow the correct derivation
# when parsing syntactically valid Python code. Since testlist_safe was created
# specifically to handle list comprehensions and generator expressions enclosed
# with parentheses, it's safe to only use it in those. That avoids the issue; we
# can parse code like set(x for x in [],).
#
# The syntax supported by this set of rules is not a valid Python 3 syntax,
# hence the prefix "old".
#
# See https://bugs.python.org/issue27494
old_comp_iter: old_comp_for | old_comp_if
old_comp_for: [ASYNC] 'for' exprlist 'in' testlist_safe [old_comp_iter]
old_comp_if: 'if' old_test [old_comp_iter]
```

testlist1: test (',' test)*

not used in grammar, but may appear in "node" passed from Parser to Compiler
encoding_decl: NAME

yield_expr: 'yield' [yield_arg]
yield_arg: 'from' test | testlist_star_expr

3.10 match statement definition

PS: normally the grammar is much much more restricted, but
at this moment for not trying to bother much with encoding the
exact same DSL in a LL(1) parser, we will just accept an expression
and let the ast.parse() step of the safe mode to reject invalid
grammar.

The reason why it is more restricted is that, patterns are some
sort of a DSL (more advanced than our LHS on assignments, but
still in a very limited python subset). They are not really
expressions, but who cares. If we can parse them, that is enough
to reformat them.

match_stmt: "match" subject_expr ':' NEWLINE INDENT case_block+ DEDENT

This is more permissive than the actual version. For example it
accepts `match *something:`, even though single-item starred expressions
are forbidden.
subject_expr: (namedexpr_test|star_expr) (',' (namedexpr_test|star_expr))* [',']

cases
case_block: "case" patterns [guard] ':' suite
guard: 'if' namedexpr_test
patterns: pattern (',' pattern)* [',']
pattern: (expr|star_expr) ['as' expr]
Empty configuration file; used in tests to avoid interference from Black's own config.
Contributing to _Black_

Welcome! Happy to see you willing to make the project better. Have you read the entire
[user documentation](https://black.readthedocs.io/en/latest/) yet?

Our [contributing documentation](https://black.readthedocs.org/en/latest/contributing/)
contains details on all you need to know about contributing to _Black_, the basics to
the internals of _Black_.

We look forward to your contributions!

```
name: Fuzz

on: [push, pull_request]

concurrency:
  group: ${{ github.workflow }}-${{ github.event.pull_request.number || github.sha }}
  cancel-in-progress: true

permissions:
  contents: read

jobs:
  build:
    # We want to run on external PRs, but not on our own internal PRs as they'll be run
    # by the push to the branch. Without this if check, checks are duplicated since
    # internal PRs match both the push and pull_request events.
    if:
      github.event_name == 'push' || github.event.pull_request.head.repo.full_name !=
      github.repository

    runs-on: ubuntu-latest
    strategy:
      fail-fast: false
      matrix:
        python-version: ["3.7", "3.8", "3.9", "3.10", "3.11"]

    steps:
      - uses: actions/checkout@v3

      - name: Set up Python ${{ matrix.python-version }}
        uses: actions/setup-python@v4
        with:
          python-version: ${{ matrix.python-version }}

      - name: Install dependencies
        run: |
          python -m pip install --upgrade pip
          python -m pip install --upgrade tox

      - name: Run fuzz tests
        run: |
          tox -e fuzz
```
Using _Black_ with other tools

Black compatible configurations

All of Black's changes are harmless (or at least, they should be), but a few do conflict
against other tools. It is not uncommon to be using other tools alongside _Black_ like

linters and type checkers. Some of them need a bit of tweaking to resolve the conflicts. Listed below are _Black_ compatible configurations in various formats for the common tools out there.

Please note that _Black_ only supports the TOML file format for its configuration (e.g. `pyproject.toml`). The provided examples are to only configure their corresponding tools, using **their** supported file formats.

Compatible configuration files can be [found here](https://github.com/psf/black/blob/main/docs/compatible_configs/).

isort

[isort](https://pypi.org/p/isort/) helps to sort and format imports in your Python code. _Black_ also formats imports, but in a different way from isort's defaults which leads to conflicting changes.

Profile

Since version 5.0.0, isort supports [profiles](https://pycqa.github.io/isort/docs/configuration/profiles.html) to allow easy interoperability with common code styles. You can set the black profile in any of the [config files](https://pycqa.github.io/isort/docs/configuration/config_files.html) supported by isort. Below, an example for `pyproject.toml`:

```toml
[tool.isort]
profile = "black"
```

Custom Configuration

If you're using an isort version that is older than 5.0.0 or you have some custom configuration for _Black_, you can tweak your isort configuration to make it compatible with _Black_. Below, an example for `.isort.cfg`:

```
multi_line_output = 3
include_trailing_comma = True
force_grid_wrap = 0
use_parentheses = True
ensure_newline_before_comments = True
line_length = 88
```

Why those options above?

Black wraps imports that surpass `line-length` by moving identifiers into their own

indented line. If that still doesn't fit the bill, it will put all of them in separate
lines and put a trailing comma. A more detailed explanation of this behaviour can be
[found here](../the_black_code_style/current_style.md#how-black-wraps-lines).

isort's default mode of wrapping imports that extend past the `line_length` limit is
"Grid".

```py3
from third_party import (lib1, lib2, lib3,
                lib4, lib5, ...)
```

This style is incompatible with _Black_, but isort can be configured to use a different
wrapping mode called "Vertical Hanging Indent" which looks like this:

```py3
from third_party import (
   lib1,
   lib2,
   lib3,
   lib4,
)
```

This style is _Black_ compatible and can be achieved by `multi-line-output = 3`. Also,
as mentioned above, when wrapping long imports _Black_ puts a trailing comma and uses
parentheses. isort should follow the same behaviour and passing the options
`include_trailing_comma = True` and `use_parentheses = True` configures that.

The option `force_grid_wrap = 0` is just to tell isort to only wrap imports that surpass
the `line_length` limit.

Finally, isort should be told to wrap imports when they surpass _Black_'s default limit
of 88 characters via `line_length = 88` as well as
`ensure_newline_before_comments = True` to ensure spacing import sections with comments
works the same as with _Black_.

Please note `ensure_newline_before_comments = True` only works since isort >= 5 but
does not break older versions so you can keep it if you are running previous versions.

Formats

<details>
<summary>.isort.cfg</summary>

```ini
[settings]
profile = black
```

```
```

</details>

<details>
<summary>setup.cfg</summary>

```ini
[isort]
profile = black
```

</details>

<details>
<summary>pyproject.toml</summary>

```toml
[tool.isort]
profile = 'black'
```

</details>

<details>
<summary>.editorconfig</summary>

```ini
[*.py]
profile = black
```

</details>

Flake8

[Flake8](https://pypi.org/p/flake8/) is a code linter. It warns you of syntax errors,
possible bugs, stylistic errors, etc. For the most part, Flake8 follows
[PEP 8](https://www.python.org/dev/peps/pep-0008/) when warning about stylistic errors.
There are a few deviations that cause incompatibilities with _Black_.

Configuration

```
max-line-length = 88
extend-ignore = E203
```

Why those options above?

In some cases, as determined by PEP 8, _Black_ will enforce an equal amount of
whitespace around slice operators. Due to this, Flake8 will raise
`E203 whitespace before ':'` warnings. Since this warning is not PEP 8 compliant, Flake8
should be configured to ignore it via `extend-ignore = E203`.

When breaking a line, _Black_ will break it before a binary operator. This is compliant
with PEP 8 as of
[April 2016](https://github.com/python/peps/commit/c59c4376ad233a62ca4b3a6060c81368bd21e85b#diff-
64ec08cc46db7540f18f2af46037f599).
There's a disabled-by-default warning in Flake8 which goes against this PEP 8
recommendation called `W503 line break before binary operator`. It should not be enabled
in your configuration.

Also, as like with isort, flake8 should be configured to allow lines up to the length
limit of `88`, _Black_'s default. This explains `max-line-length = 88`.

Formats

<details>
<summary>.flake8</summary>

```ini
[flake8]
max-line-length = 88
extend-ignore = E203
```

</details>

<details>
<summary>setup.cfg</summary>

```ini
[flake8]
max-line-length = 88
extend-ignore = E203
```

</details>

<details>
<summary>tox.ini</summary>

```ini
[flake8]
max-line-length = 88
```

extend-ignore = E203
```

</details>

Pylint

[Pylint](https://pypi.org/p/pylint/) is also a code linter like Flake8. It has the same
checks as flake8 and more. In particular, it has more formatting checks regarding style
conventions like variable naming. With so many checks, Pylint is bound to have some
mixed feelings about _Black_'s formatting style.

Configuration

```
max-line-length = 88
```

Why those options above?

Pylint should be configured to only complain about lines that surpass `88` characters
via `max-line-length = 88`.

If using `pylint<2.6.0`, also disable `C0326` and `C0330` as these are incompatible with
Black formatting and have since been removed.

Formats

<details>
<summary>pylintrc</summary>

```ini
[format]
max-line-length = 88
```

</details>

<details>
<summary>setup.cfg</summary>

```cfg
[pylint]
max-line-length = 88
```

</details>

<details>
<summary>pyproject.toml</summary>

```toml
[tool.pylint.format]
max-line-length = "88"
```

</details>

pycodestyle

[pycodestyle](https://pycodestyle.pycqa.org/) is also a code linter like Flake8.

Configuration

```
max-line-length = 88
ignore = E203
```

Why those options above?

pycodestyle should be configured to only complain about lines that surpass `88`
characters via `max_line_length = 88`.

See
[Why are Flake8s E203 and W503 violated?](https://black.readthedocs.io/en/stable/faq.html#why-are-flake8-s-e203-and-w503-violated)

Formats

<details>
<summary>setup.cfg</summary>

```cfg
[pycodestyle]
ignore = E203
max_line_length = 88
```

</details>

The basics

Foundational knowledge on using and configuring Black.

Black is a well-behaved Unix-style command-line tool:

- it does nothing if it finds no sources to format;
- it will read from standard input and write to standard output if `-` is used as the
 filename;
- it only outputs messages to users on standard error;
- exits with code 0 unless an internal error occurred or a CLI option prompted it.

Usage

To get started right away with sensible defaults:

```sh
black {source_file_or_directory}
```

You can run _Black_ as a package if running it as a script doesn't work:

```sh
python -m black {source_file_or_directory}
```

Command line options

The CLI options of _Black_ can be displayed by expanding the view below or by running
`black --help`. While _Black_ has quite a few knobs these days, it is still opinionated
so style options are deliberately limited and rarely added.

<details>

<summary>CLI reference</summary>

```{program-output} black --help

```

</details>

Note that all command-line options listed above can also be configured using a
`pyproject.toml` file (more on that below).

Code input alternatives

Standard Input

Black supports formatting code via stdin, with the result being printed to stdout.
Just let _Black_ know with `-` as the path.

```console
$ echo "print ( 'hello, world' )" | black -
```

```
print("hello, world")
reformatted -
All done!
1 file reformatted.
```

Tip: if you need _Black_ to treat stdin input as a file passed directly via the CLI,
use `--stdin-filename`. Useful to make sure _Black_ will respect the `--force-exclude`
option on some editors that rely on using stdin.

As a string

You can also pass code as a string using the `-c` / `--code` option.

```console
$ black --code "print ( 'hello, world' )"
print("hello, world")
```

Writeback and reporting

By default _Black_ reformats the files given and/or found in place. Sometimes you need
Black to just tell you what it _would_ do without actually rewriting the Python files.

There's two variations to this mode that are independently enabled by their respective
flags. Both variations can be enabled at once.

(labels/exit-code)=

Exit code

Passing `--check` will make _Black_ exit with:

- code 0 if nothing would change;
- code 1 if some files would be reformatted; or
- code 123 if there was an internal error

```console
$ black test.py --check
All done!
1 file would be left unchanged.
$ echo $?
0

$ black test.py --check
would reformat test.py
Oh no!
1 file would be reformatted.
```

```
$ echo $?
1
```

```
$ black test.py --check
error: cannot format test.py: INTERNAL ERROR: Black produced code that is not equivalent to the source.  Please
report a bug on https://github.com/psf/black/issues.  This diff might be helpful: /tmp/blk_kjdr1oog.log
Oh no!
1 file would fail to reformat.
$ echo $?
123
```

Diffs

Passing `--diff` will make _Black_ print out diffs that indicate what changes _Black_
would've made. They are printed to stdout so capturing them is simple.

If you'd like colored diffs, you can enable them with the `--color`.

```console
$ black test.py --diff
--- test.py     2021-03-08 22:23:40.848954 +0000
+++ test.py     2021-03-08 22:23:47.126319 +0000
@@ -1 +1 @@
-print ( 'hello, world' )
+print("hello, world")
would reformat test.py
All done!
1 file would be reformatted.
```

Output verbosity

Black in general tries to produce the right amount of output, balancing between
usefulness and conciseness. By default, _Black_ emits files modified and error messages,
plus a short summary.

```console
$ black src/
error: cannot format src/black_primer/cli.py: Cannot parse: 5:6: mport asyncio
reformatted src/black_primer/lib.py
reformatted src/blackd/__init__.py
reformatted src/black/__init__.py
Oh no!
3 files reformatted, 2 files left unchanged, 1 file failed to reformat.
```

Passing `-v` / `--verbose` will cause _Black_ to also emit messages about files that

were not changed or were ignored due to exclusion patterns. If _Black_ is using a
configuration file, a blue message detailing which one it is using will be emitted.

```console
$ black src/ -v
Using configuration from /tmp/pyproject.toml.
src/blib2to3 ignored: matches the --extend-exclude regular expression
src/_black_version.py wasn't modified on disk since last run.
src/black/__main__.py wasn't modified on disk since last run.
error: cannot format src/black_primer/cli.py: Cannot parse: 5:6: mport asyncio
reformatted src/black_primer/lib.py
reformatted src/blackd/__init__.py
reformatted src/black/__init__.py
Oh no!
3 files reformatted, 2 files left unchanged, 1 file failed to reformat
```

Passing `-q` / `--quiet` will cause _Black_ to stop emitting all non-critial output.
Error messages will still be emitted (which can silenced by `2>/dev/null`).

```console
$ black src/ -q
error: cannot format src/black_primer/cli.py: Cannot parse: 5:6: mport asyncio
```

Versions

You can check the version of _Black_ you have installed using the `--version` flag.

```console
$ black --version
black, version 23.1.0
```

An option to require a specific version to be running is also provided.

```console
$ black --required-version 21.9b0 -c "format = 'this'"
format = "this"
$ black --required-version 31.5b2 -c "still = 'beta?!'"
Oh no!   The required version does not match the running version!
```

This is useful for example when running _Black_ in multiple environments that haven't
necessarily installed the correct version. This option can be set in a configuration
file for consistent results across environments.

Configuration via a file

Black is able to read project-specific default values for its command line options from a `pyproject.toml` file. This is especially useful for specifying custom `--include` and `--exclude`/`--force-exclude`/`--extend-exclude` patterns for your project.

Pro-tip: If you're asking yourself "Do I need to configure anything?" the answer is "No". _Black_ is all about sensible defaults. Applying those defaults will have your code in compliance with many other _Black_ formatted projects.

What on Earth is a `pyproject.toml` file?

[PEP 518](https://www.python.org/dev/peps/pep-0518/) defines `pyproject.toml` as a configuration file to store build system requirements for Python projects. With the help of tools like [Poetry](https://python-poetry.org/), [Flit](https://flit.readthedocs.io/en/latest/), or [Hatch](https://hatch.pypa.io/latest/) it can fully replace the need for `setup.py` and `setup.cfg` files.

Where _Black_ looks for the file

By default _Black_ looks for `pyproject.toml` starting from the common base directory of all files and directories passed on the command line. If it's not there, it looks in parent directories. It stops looking when it finds the file, or a `.git` directory, or a `.hg` directory, or the root of the file system, whichever comes first.

If you're formatting standard input, _Black_ will look for configuration starting from the current working directory.

You can use a "global" configuration, stored in a specific location in your home directory. This will be used as a fallback configuration, that is, it will be used if and only if _Black_ doesn't find any configuration as mentioned above. Depending on your operating system, this configuration file should be stored as:

- Windows: `~\.black`
- Unix-like (Linux, MacOS, etc.): `$XDG_CONFIG_HOME/black` (`~/.config/black` if the `XDG_CONFIG_HOME` environment variable is not set)

Note that these are paths to the TOML file itself (meaning that they shouldn't be named as `pyproject.toml`), not directories where you store the configuration. Here, `~` refers to the path to your home directory. On Windows, this will be something like `C:\\Users\UserName`.

You can also explicitly specify the path to a particular file that you want with `--config`. In this situation _Black_ will not look for any other file.

If you're running with `--verbose`, you will see a blue message if a file was found and used.

Please note `blackd` will not use `pyproject.toml` configuration.

Configuration format

As the file extension suggests, `pyproject.toml` is a
[TOML](https://github.com/toml-lang/toml) file. It contains separate sections for
different tools. _Black_ is using the `[tool.black]` section. The option keys are the
same as long names of options on the command line.

Note that you have to use single-quoted strings in TOML for regular expressions. It's
the equivalent of r-strings in Python. Multiline strings are treated as verbose regular
expressions by Black. Use `[ ]` to denote a significant space character.

<details>
<summary>Example <code>pyproject.toml</code></summary>

```toml
[tool.black]
line-length = 88
target-version = ['py37']
include = '\.pyi?$'
# 'extend-exclude' excludes files or directories in addition to the defaults
extend-exclude = '''
# A regex preceded with ^/ will apply only to files and directories
# in the root of the project.
(
  ^/foo.py    # exclude a file named foo.py in the root of the project
  | .*_pb2.py  # exclude autogenerated Protocol Buffer files anywhere in the project
)
'''
```

</details>

Lookup hierarchy

Command-line options have defaults that you can see in `--help`. A `pyproject.toml` can
override those defaults. Finally, options provided by the user on the command line
override both.

Black will only ever use one `pyproject.toml` file during an entire run. It doesn't
look for multiple files, and doesn't compose configuration from different levels of the
file hierarchy.

Next steps

A good next step would be configuring auto-discovery so `black .` is all you need

instead of laborously listing every file or directory. You can get started by heading over to [File collection and discovery](./file_collection_and_discovery.md).

Another good choice would be setting up an [integration with your editor](../integrations/editors.md) of choice or with [pre-commit for source version control](../integrations/source_version_control.md).
FROM python:3-slim AS builder

RUN mkdir /src
COPY . /src/
ENV VIRTUAL_ENV=/opt/venv
RUN python -m venv $VIRTUAL_ENV
RUN . /opt/venv/bin/activate && pip install --no-cache-dir --upgrade pip setuptools wheel \
    # Install build tools to compile dependencies that don't have prebuilt wheels
    && apt update && apt install -y git build-essential \
    && cd /src \
    && pip install --no-cache-dir .[colorama,d]

FROM python:3-slim

copy only Python packages to limit the image size
COPY --from=builder /opt/venv /opt/venv
ENV PATH="/opt/venv/bin:$PATH"

CMD ["/opt/venv/bin/black"]
Gauging changes

A lot of the time, your change will affect formatting and/or performance. Quantifying these changes is hard, so we have tooling to help make it easier.

It's recommended you evaluate the quantifiable changes your _Black_ formatting modification causes before submitting a PR. Think about if the change seems disruptive enough to cause frustration to projects that are already "black formatted".

diff-shades

diff-shades is a tool that runs _Black_ across a list of open-source projects recording the results. The main highlight feature of diff-shades is being able to compare two revisions of _Black_. This is incredibly useful as it allows us to see what exact changes will occur, say merging a certain PR.

For more information, please see the [diff-shades documentation][diff-shades].

CI integration

diff-shades is also the tool behind the "diff-shades results comparing ..." / "diff-shades reports zero changes ..." comments on PRs. The project has a GitHub Actions workflow that analyzes and compares two revisions of _Black_ according to these rules:

	Baseline revision	Target revision
On PRs	latest commit on `main`	PR commit with `main` merged
On pushes (main only)	latest PyPI version	the pushed commit

For pushes to main, there's only one analysis job named `preview-changes` where the preview style is used for all projects.

For PRs they get one more analysis job: `assert-no-changes`. It's similar to `preview-changes` but runs with the stable code style. It will fail if changes were made. This makes sure code won't be reformatted again and again within the same year in accordance to Black's stability policy.

Additionally for PRs, a PR comment will be posted embedding a summary of the preview changes and links to further information. If there's a pre-existing diff-shades comment, it'll be updated instead the next time the workflow is triggered on the same PR.

```{note}
The `preview-changes` job will only fail intentionally if while analyzing a file failed to format. Otherwise a failure indicates a bug in the workflow.
```

The workflow uploads several artifacts upon completion:

- The raw analyses (.json)
- HTML diffs (.html)
- `.pr-comment.json` (if triggered by a PR)

The last one is downloaded by the `diff-shades-comment` workflow and shouldn't be downloaded locally. The HTML diffs come in handy for push-based where there's no PR to post a comment. And the analyses exist just in case you want to do further analysis using the collected data locally.

[diff-shades]: https://github.com/ichard26/diff-shades#readme
Note: don't use this config for your own repositories. Instead, see
"Version control integration" in docs/integrations/source_version_control.md
exclude: ^(src/blib2to3/|profiling/|tests/data/)
repos:
 - repo: local
   hooks:
     - id: check-pre-commit-rev-in-example
       name: Check pre-commit rev in example
       language: python
       entry: python -m scripts.check_pre_commit_rev_in_example
       files: '(CHANGES\.md|source_version_control\.md)$'
       additional_dependencies:
         &version_check_dependencies [

```
          commonmark==0.9.1,
          pyyaml==5.4.1,
          beautifulsoup4==4.9.3,
        ]

    - id: check-version-in-the-basics-example
      name: Check black version in the basics example
      language: python
      entry: python -m scripts.check_version_in_basics_example
      files: '(CHANGES\.md|the_basics\.md)$'
      additional_dependencies: *version_check_dependencies

- repo: https://github.com/pycqa/isort
  rev: 5.12.0
  hooks:
    - id: isort

- repo: https://github.com/pycqa/flake8
  rev: 4.0.1
  hooks:
    - id: flake8
      additional_dependencies:
        - flake8-bugbear
        - flake8-comprehensions
        - flake8-simplify

- repo: https://github.com/pre-commit/mirrors-mypy
  rev: v0.991
  hooks:
    - id: mypy
      exclude: ^docs/conf.py
      additional_dependencies:
        - types-dataclasses >= 0.1.3
        - types-PyYAML
        - tomli >= 0.2.6, < 2.0.0
        - types-typed-ast >= 1.4.1
        - click >= 8.1.0
        - packaging >= 22.0
        - platformdirs >= 2.1.0
        - pytest
        - hypothesis

- repo: https://github.com/pre-commit/mirrors-prettier
  rev: v2.7.1
  hooks:
    - id: prettier
      exclude: \.github/workflows/diff_shades\.yml
```

```
  - repo: https://github.com/pre-commit/pre-commit-hooks
    rev: v4.3.0
    hooks:
      - id: end-of-file-fixer
      - id: trailing-whitespace
```
Release process

Black has had a lot of work done into standardizing and automating its release
process. This document sets out to explain how everything works and how to release
Black using said automation.

Release cadence

We aim to release whatever is on `main` every 1-2 months. This ensures merged
improvements and bugfixes are shipped to users reasonably quickly, while not massively
fracturing the user-base with too many versions. This also keeps the workload on
maintainers consistent and predictable.

If there's not much new on `main` to justify a release, it's acceptable to skip a
month's release. Ideally January releases should not be skipped because as per our
[stability policy](labels/stability-policy), the first release in a new calendar year
may make changes to the _stable_ style. While the policy applies to the first release
(instead of only January releases), confining changes to the stable style to January
will keep things predictable (and nicer) for users.

Unless there is a serious regression or bug that requires immediate patching, **there
should not be more than one release per month**. While version numbers are cheap,
releases require a maintainer to both commit to do the actual cutting of a release, but
also to be able to deal with the potential fallout post-release. Releasing more
frequently than monthly nets rapidly diminishing returns.

Cutting a release

You must have `write` permissions for the _Black_ repository to cut a release.

The 10,000 foot view of the release process is that you prepare a release PR and then
publish a [GitHub Release]. This triggers [release automation](#release-workflows) that
builds all release artifacts and publishes them to the various platforms we publish to.

To cut a release:

1. Determine the release's version number
   - **_Black_ follows the [CalVer] versioning standard using the `YY.M.N` format**
     - So unless there already has been a release during this month, `N` should be `0`
   - Example: the first release in January, 2022 `22.1.0`
1. File a PR editing `CHANGES.md` and the docs to version the latest changes
   1. Replace the `## Unreleased` header with the version number
   1. Remove any empty sections for the current release

1. (_optional_) Read through and copy-edit the changelog (eg. by moving entries, fixing typos, or rephrasing entries)
1. Add a new empty template for the next release above ([template below](#changelog-template))
1. Update references to the latest version in {doc}`/integrations/source_version_control` and {doc}`/usage_and_configuration/the_basics`
- Example PR: [GH-3139]
1. Once the release PR is merged, wait until all CI passes
- If CI does not pass, **stop** and investigate the failure(s) as generally we'd want to fix failing CI before cutting a release
1. [Draft a new GitHub Release][new-release]
    1. Click `Choose a tag` and type in the version number, then select the `Create new tag: YY.M.N on publish` option that appears
    1. Verify that the new tag targets the `main` branch
    1. You can leave the release title blank, GitHub will default to the tag name
    1. Copy and paste the _raw changelog Markdown_ for the current release into the description box
1. Publish the GitHub Release, triggering [release automation](#release-workflows) that will handle the rest
1. At this point, you're basically done. It's good practice to go and [watch and verify that all the release workflows pass][black-actions], although you will receive a GitHub notification should something fail.
   - If something fails, don't panic. Please go read the respective workflow's logs and configuration file to reverse-engineer your way to a fix/solution.

Congratulations! You've successfully cut a new release of _Black_. Go and stand up and take a break, you deserve it.

```{important}
Once the release artifacts reach PyPI, you may see new issues being filed indicating regressions. While regressions are not great, they don't automatically mean a hotfix release is warranted. Unless the regressions are serious and impact many users, a hotfix release is probably unnecessary.

In the end, use your best judgement and ask other maintainers for their thoughts.
```

Changelog template

Use the following template for a clean changelog after the release:

```
## Unreleased

### Highlights

<!-- Include any especially major or disruptive changes here -->
```

Stable style

<!-- Changes that affect Black's stable style -->

Preview style

<!-- Changes that affect Black's preview style -->

Configuration

<!-- Changes to how Black can be configured -->

Packaging

<!-- Changes to how Black is packaged, such as dependency requirements -->

Parser

<!-- Changes to the parser or to version autodetection -->

Performance

<!-- Changes that improve Black's performance. -->

Output

<!-- Changes to Black's terminal output and error messages -->

Blackd

<!-- Changes to blackd -->

Integrations

<!-- For example, Docker, GitHub Actions, pre-commit, editors -->

Documentation

<!-- Major changes to documentation and policies. Small docs changes
   don't need a changelog entry. -->
```

## Release workflows

All of _Black_'s release automation uses [GitHub Actions]. All workflows are therefore
configured using YAML files in the `.github/workflows` directory of the _Black_
repository.

---

They are triggered by the publication of a [GitHub Release].

Below are descriptions of our release workflows.

### Publish to PyPI

This is our main workflow. It builds an [sdist] and [wheels] to upload to PyPI where the vast majority of users will download Black from. It's divided into three job groups:

#### sdist + pure wheel

This single job builds the sdist and pure Python wheel (i.e., a wheel that only contains Python code) using [build] and then uploads them to PyPI using [twine]. These artifacts are general-purpose and can be used on basically any platform supported by Python.

#### mypyc wheels ()

We use [mypyc] to compile _Black_ into a CPython C extension for significantly improved performance. Wheels built with mypyc are platform and Python version specific. [Supported platforms are documented in the FAQ](labels/mypyc-support).

These matrix jobs use [cibuildwheel] which handles the complicated task of building C extensions for many environments for us. Since building these wheels is slow, there are multiple mypyc wheels jobs (hence the term "matrix") that build for a specific platform (as noted in the job name in parentheses).

Like the previous job group, the built wheels are uploaded to PyPI using [twine].

#### Update stable branch

So this job doesn't _really_ belong here, but updating the `stable` branch after the other PyPI jobs pass (they must pass for this job to start) makes the most sense. This saves us from remembering to update the branch sometime after cutting the release.

- _Currently this workflow uses an API token associated with @ambv's PyPI account_

### Publish executables

This workflow builds native executables for multiple platforms using [PyInstaller]. This allows people to download the executable for their platform and run _Black_ without a [Python runtime](https://wiki.python.org/moin/PythonImplementations) installed.

The created binaries are stored on the associated GitHub Release for download over _IPv4 only_ (GitHub still does not have IPv6 access ).

### docker

This workflow uses the QEMU powered `buildx` feature of Docker to upload an `arm64` and
`amd64`/`x86_64` build of the official _Black_ Docker image.

- _Currently this workflow uses an API Token associated with @cooperlees account_

```{note}
This also runs on each push to `main`.
```

[black-actions]: https://github.com/psf/black/actions
[build]: https://pypa-build.readthedocs.io/
[calver]: https://calver.org
[cibuildwheel]: https://cibuildwheel.readthedocs.io/
[gh-3139]: https://github.com/psf/black/pull/3139
[github actions]: https://github.com/features/actions
[github release]: https://github.com/psf/black/releases
[new-release]: https://github.com/psf/black/releases/new
[mypyc]: https://mypyc.readthedocs.io/
[mypyc-platform-support]:
 /faq.html#what-is-compiled-yes-no-all-about-in-the-version-output
[pyinstaller]: https://www.pyinstaller.org/
[sdist]:
 https://packaging.python.org/en/latest/glossary/#term-Source-Distribution-or-sdist
[twine]: https://github.com/features/actions
[wheels]: https://packaging.python.org/en/latest/glossary/#term-Wheel
Guides

```{toctree}
---
hidden:
---

introducing_black_to_your_project
using_black_with_other_tools
```

Wondering how to do something specific? You've found the right place! Listed below are
topic specific guides available:

- {doc}`introducing_black_to_your_project`
- {doc}`using_black_with_other_tools`
--- [Deterministic header]
+++ [Deterministic header]
@@ -1,8 +1,8 @@
...
-'some_string'
-b'\\xa3'
+"some_string"

+b"\\xa3"

Name

None

True

False

1

@@ -21,99 +21,118 @@

Name1 or (Name2 and Name3) or Name4

Name1 or Name2 and Name3 or Name4

v1 << 2

1 >> v2

1 % finished

-1 + v2 - v3 * 4 ^ 5 ** v6 / 7 // 8

-((1 + v2) - (v3 * 4)) ^ (((5 ** v6) / 7) // 8)

+1 + v2 - v3 * 4 ^ 5**v6 / 7 // 8

+((1 + v2) - (v3 * 4)) ^ (((5**v6) / 7) // 8)

not great

~great

+value

-1

~int and not v1 ^ 123 + v2 | True

(~int) and (not ((v1 ^ (123 + v2)) | True))

-+really ** -confusing ** ~operator ** -precedence

-flags & ~ select.EPOLLIN and waiters.write_task is not None

++(really ** -(confusing ** ~(operator**-precedence)))

+flags & ~select.EPOLLIN and waiters.write_task is not None

lambda arg: None

lambda a=True: a

lambda a, b, c=True: a

-lambda a, b, c=True, *, d=(1 << v2), e='str': a

-lambda a, b, c=True, *vararg, d=(v1 << 2), e='str', **kwargs: a + b

+lambda a, b, c=True, *, d=(1 << v2), e="str": a

+lambda a, b, c=True, *vararg, d=(v1 << 2), e="str", **kwargs: a + b

manylambdas = lambda x=lambda y=lambda z=1: z: y(): x()

-foo = (lambda port_id, ignore_missing: {"port1": port1_resource, "port2": port2_resource}[port_id])

+foo = lambda port_id, ignore_missing: {

+   "port1": port1_resource,

+   "port2": port2_resource,

+}[port_id]

1 if True else 2

str or None if True else str or bytes or None

(str or None) if True else (str or bytes or None)

str or None if (1 if True else 2) else str or bytes or None

(str or None) if (1 if True else 2) else (str or bytes or None)

-((super_long_variable_name or None) if (1 if super_long_test_name else 2) else (str or bytes or None))

-{'2.7': dead, '3.7': (long_live or die_hard)}

-{'2.7': dead, '3.7': (long_live or die_hard), **{'3.6': verygood}}

+(

```
+    (super_long_variable_name or None)
+    if (1 if super_long_test_name else 2)
+    else (str or bytes or None)
+)
+{"2.7": dead, "3.7": (long_live or die_hard)}
+{"2.7": dead, "3.7": (long_live or die_hard), **{"3.6": verygood}}
{**a, **b, **c}
-{'2.7', '3.6', '3.7', '3.8', '3.9', ('4.0' if gilectomy else '3.10')}
-({'a': 'b'}, (True or False), (+value), 'string', b'bytes') or None
+{"2.7", "3.6", "3.7", "3.8", "3.9", ("4.0" if gilectomy else "3.10")}
+({"a": "b"}, (True or False), (+value), "string", b"bytes") or None
()
(1,)
(1, 2)
(1, 2, 3)
[]
[1, 2, 3, 4, 5, 6, 7, 8, 9, (10 or A), (11 or B), (12 or C)]
-[1, 2, 3,]
+[1, 2, 3]
[*a]
[*range(10)]
-[*a, 4, 5,]
-[4, *a, 5,]
-[this_is_a_very_long_variable_which_will_force_a_delimiter_split, element, another, *more]
+[*a, 4, 5]
+[4, *a, 5]
+[
+    this_is_a_very_long_variable_which_will_force_a_delimiter_split,
+    element,
+    another,
+    *more,
+]
{i for i in (1, 2, 3)}
-{(i ** 2) for i in (1, 2, 3)}
-{(i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c'))}
-{((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3)}
+{(i**2) for i in (1, 2, 3)}
+{(i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c"))}
+{((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3)}
[i for i in (1, 2, 3)]
-[(i ** 2) for i in (1, 2, 3)]
-[(i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c'))]
-[((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3)]
+[(i**2) for i in (1, 2, 3)]
+[(i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c"))]
+[((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3)]
{i: 0 for i in (1, 2, 3)}
-{i: j for i, j in ((1, 'a'), (2, 'b'), (3, 'c'))}
```

```
+{i: j for i, j in ((1, "a"), (2, "b"), (3, "c"))}
{a: b * 2 for a, b in dictionary.items()}
{a: b * -2 for a, b in dictionary.items()}
-{k: v for k, v in
this_is_a_very_long_variable_which_will_cause_a_trailing_comma_which_breaks_the_comprehension}
+{
+    k: v
+    for k, v in
this_is_a_very_long_variable_which_will_cause_a_trailing_comma_which_breaks_the_comprehension
+}
Python3 > Python2 > COBOL
Life is Life
call()
call(arg)
-call(kwarg='hey')
-call(arg, kwarg='hey')
-call(arg, another, kwarg='hey', **kwargs)
-call(this_is_a_very_long_variable_which_will_force_a_delimiter_split, arg, another, kwarg='hey', **kwargs)  #
note: no trailing comma pre-3.6
+call(kwarg="hey")
+call(arg, kwarg="hey")
+call(arg, another, kwarg="hey", **kwargs)
+call(
+    this_is_a_very_long_variable_which_will_force_a_delimiter_split,
+    arg,
+    another,
+    kwarg="hey",
+    **kwargs
+)  # note: no trailing comma pre-3.6
call(*gidgets[:2])
call(a, *gidgets[:2])
call(**self.screen_kwargs)
call(b, **self.screen_kwargs)
lukasz.langa.pl
call.me(maybe)
-1 .real
-1.0 .real
+(1).real
+(1.0).real
....__class__
list[str]
dict[str, int]
tuple[str, ...]
-tuple[
-    str, int, float, dict[str, int]
-]
-tuple[str, int, float, dict[str, int],]
+tuple[str, int, float, dict[str, int]]
```

```
+tuple[str, int, float, dict[str, int]]
very_long_variable_name_filters: t.List[
    t.Tuple[str, t.Union[str, t.List[t.Optional[str]]]],
]
xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(  # type: ignore
    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
)
xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(  # type: ignore
    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
)
-xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[
-    ..., List[SomeClass]
-] = classmethod(sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__))  # type: ignore
+xxxx_xxx_xxxx_xxxxx_xxxx_xxx: Callable[..., List[SomeClass]] = classmethod(
+    sync(async_xxxx_xxx_xxxx_xxxxx_xxxx_xxx.__func__)
+)  # type: ignore
slice[0]
slice[0:1]
slice[0:1:2]
slice[:]
slice[:-1]
@@ -137,118 +156,197 @@
numpy[-(c + 1) :, d]
numpy[:, l[-2]]
numpy[:, ::-1]
numpy[np.newaxis, :]
(str or None) if (sys.version_info[0] > (3,)) else (str or bytes or None)
-{'2.7': dead, '3.7': long_live or die_hard}
-{'2.7', '3.6', '3.7', '3.8', '3.9', '4.0' if gilectomy else '3.10'}
+{"2.7": dead, "3.7": long_live or die_hard}
+{"2.7", "3.6", "3.7", "3.8", "3.9", "4.0" if gilectomy else "3.10"}
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10 or A, 11 or B, 12 or C]
(SomeName)
SomeName
(Good, Bad, Ugly)
(i for i in (1, 2, 3))
-((i ** 2) for i in (1, 2, 3))
-((i ** 2) for i, _ in ((1, 'a'), (2, 'b'), (3, 'c')))
-(((i ** 2) + j) for i in (1, 2, 3) for j in (1, 2, 3))
+((i**2) for i in (1, 2, 3))
+((i**2) for i, _ in ((1, "a"), (2, "b"), (3, "c")))
+(((i**2) + j) for i in (1, 2, 3) for j in (1, 2, 3))
(*starred,)
-{"id": "1","type": "type","started_at": now(),"ended_at": now() + timedelta(days=10),"priority":
1,"import_session_id": 1,**kwargs}
+{
+    "id": "1",
+    "type": "type",
```

```
+    "started_at": now(),
+    "ended_at": now() + timedelta(days=10),
+    "priority": 1,
+    "import_session_id": 1,
+    **kwargs,
+}
a = (1,)
-b = 1,
+b = (1,)
c = 1
d = (1,) + a + (2,)
e = (1,).count(1)
f = 1, *range(10)
g = 1, *"ten"
-what_is_up_with_those_new_coord_names = (coord_names + set(vars_to_create)) + set(vars_to_remove)
-what_is_up_with_those_new_coord_names = (coord_names | set(vars_to_create)) - set(vars_to_remove)
-result = session.query(models.Customer.id).filter(models.Customer.account_id == account_id,
models.Customer.email == email_address).order_by(models.Customer.id.asc()).all()
-result = session.query(models.Customer.id).filter(models.Customer.account_id == account_id,
models.Customer.email == email_address).order_by(models.Customer.id.asc(),).all()
+what_is_up_with_those_new_coord_names = (coord_names + set(vars_to_create)) + set(
+    vars_to_remove
+)
+what_is_up_with_those_new_coord_names = (coord_names | set(vars_to_create)) - set(
+    vars_to_remove
+)
+result = (
+    session.query(models.Customer.id)
+    .filter(
+        models.Customer.account_id == account_id, models.Customer.email == email_address
+    )
+    .order_by(models.Customer.id.asc())
+    .all()
+)
+result = (
+    session.query(models.Customer.id)
+    .filter(
+        models.Customer.account_id == account_id, models.Customer.email == email_address
+    )
+    .order_by(models.Customer.id.asc())
+    .all()
+)
 = set()
authors.ukasz.say_thanks()
mapping = {
    A: 0.25 * (10.0 / 12),
    B: 0.1 * (10.0 / 12),
    C: 0.1 * (10.0 / 12),
```

```
      D: 0.1 * (10.0 / 12),
}


+
def gen():
    yield from outside_of_generator
-    a = (yield)
-    b = ((yield))
-    c = (((yield)))
+    a = yield
+    b = yield
+    c = yield
+


async def f():
    await some.complicated[0].call(with_args=(True or (1 is not 1)))
-print(* [] or [1])
+
+
+print(*[] or [1])
print(**{1: 3} if False else {x: x for x in range(3)})
-print(* lambda x: x)
-assert(not Test),("Short message")
-assert this is ComplexTest and not requirements.fit_in_a_single_line(force=False), "Short message"
-assert(((parens is TooMany)))
-for x, in (1,), (2,), (3,): ...
-for y in (): ...
-for z in (i for i in (1, 2, 3)): ...
-for i in (call()): ...
-for j in (1 + (2 + 3)): ...
-while(this and that): ...
-for addr_family, addr_type, addr_proto, addr_canonname, addr_sockaddr in socket.getaddrinfo('google.com',
'http'):
+print(*lambda x: x)
+assert not Test, "Short message"
+assert this is ComplexTest and not requirements.fit_in_a_single_line(
+    force=False
+), "Short message"
+assert parens is TooMany
+for (x,) in (1,), (2,), (3,):
+    ...
+for y in ():
+    ...
+for z in (i for i in (1, 2, 3)):
+    ...
+for i in call():
+    ...
+for j in 1 + (2 + 3):
```

```
+    ...
+while this and that:
+    ...
+for (
+    addr_family,
+    addr_type,
+    addr_proto,
+    addr_canonname,
+    addr_sockaddr,
+) in socket.getaddrinfo("google.com", "http"):
    pass
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp in
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp not in
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp is
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-a = aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp is not
qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
-if (
-    threading.current_thread() != threading.main_thread() and
-    threading.current_thread() != threading.main_thread() or
-    signal.getsignal(signal.SIGINT) != signal.default_int_handler
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa |
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa &
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa +
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa -
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa *
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

```
-):
-    return True
-if (
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa /
-    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
-):
-    return True
-if (
-    ~ aaaa.a + aaaa.b - aaaa.c * aaaa.d / aaaa.e | aaaa.f & aaaa.g % aaaa.h ^ aaaa.i << aaaa.k >> aaaa.l ** aaaa.m //
aaaa.n
-):
-    return True
-if (
-    ~ aaaaaaaa.a + aaaaaaaa.b - aaaaaaaa.c @ aaaaaaaa.d / aaaaaaaa.e | aaaaaaaa.f & aaaaaaaa.g % aaaaaaaa.h ^
aaaaaaaa.i << aaaaaaaa.k >> aaaaaaaa.l ** aaaaaaaa.m // aaaaaaaa.n
-):
-    return True
-if (
-    ~ aaaaaaaaaaaaaaaa.a + aaaaaaaaaaaaaaaa.b - aaaaaaaaaaaaaaaa.c * aaaaaaaaaaaaaaaa.d @ aaaaaaaaaaaaaaaa.e |
aaaaaaaaaaaaaaaa.f & aaaaaaaaaaaaaaaa.g % aaaaaaaaaaaaaaaa.h ^ aaaaaaaaaaaaaaaa.i << aaaaaaaaaaaaaaaa.k >>
aaaaaaaaaaaaaaaa.l ** aaaaaaaaaaaaaaaa.m // aaaaaaaaaaaaaaaa.n
-):
-    return True
-aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa - aaaaaaaaaaaaaaaa * (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa) /
(aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)
+a = (
+    aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp
+    in qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
+)
+a = (
+    aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp
+    not in qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
+)
+a = (
+    aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp
+    is qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
+)
+a = (
+    aaaa.bbbb.cccc.dddd.eeee.ffff.gggg.hhhh.iiii.jjjj.kkkk.llll.mmmm.nnnn.oooo.pppp
+    is not qqqq.rrrr.ssss.tttt.uuuu.vvvv.xxxx.yyyy.zzzz
+)
+if (
+    threading.current_thread() != threading.main_thread()
+    and threading.current_thread() != threading.main_thread()
+    or signal.getsignal(signal.SIGINT) != signal.default_int_handler
+):
+    return True
+if (
```

```
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    | aaaaaaaaaaaaaaaaaaaaa*aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    & aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    + aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    - aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    * aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    / aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+):
+    return True
+if (
+    ~aaaa.a + aaaa.b - aaaa.c * aaaa.d / aaaa.e
+    | aaaa.f & aaaa.g % aaaa.h ^ aaaa.i << aaaa.k >> aaaa.l**aaaa.m // aaaa.n
+):
+    return True
+if (
+    ~aaaaaaaa.a + aaaaaaaa.b - aaaaaaaa.c @ aaaaaaaa.d / aaaaaaaa.e
+    | aaaaaaaa.f & aaaaaaaa.g % aaaaaaaa.h
+    ^ aaaaaaaa.i << aaaaaaaa.k >> aaaaaaaa.l**aaaaaaaa.m // aaaaaaaa.n
+):
+    return True
+if (
+    ~aaaaaaaaaaaaaaaa.a
+    + aaaaaaaaaaaaaaaa.b
+    - aaaaaaaaaaaaaaaa.c * aaaaaaaaaaaaaaaa.d @ aaaaaaaaaaaaaaaa.e
+    | aaaaaaaaaaaaaaaa.f & aaaaaaaaaaaaaaaa.g % aaaaaaaaaaaaaaaa.h
+    ^ aaaaaaaaaaaaaaaa.i
+    << aaaaaaaaaaaaaaaa.k
+    >> aaaaaaaaaaaaaaaa.l**aaaaaaaaaaaaaaaa.m // aaaaaaaaaaaaaaaa.n
```

```
+):
+    return True
+(
+    aaaaaaaaaaaaaaaa
+    + aaaaaaaaaaaaaaaa
+    - aaaaaaaaaaaaaaaa
+    * (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)
+    / (aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa)
+)
aaaaaaaaaaaaaaaa + aaaaaaaaaaaaaaaa
-aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa >>
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa <<
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+(
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    >> aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    << aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+)
bbbb >> bbbb * bbbb
-aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ^bbbb.a &
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa^aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaa
+(
+    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    ^ bbbb.a & aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+    ^ aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
+)
last_call()
# standalone comment at ENDMARKER
# Copyright 2006 Google, Inc. All Rights Reserved.
# Licensed to PSF under a Contributor Agreement.

# A grammar to describe tree matching patterns.
# Not shown here:
# - 'TOKEN' stands for any token (leaf node)
# - 'any' stands for any node (leaf or interior)
# With 'any' we can still specify the sub-structure.

# The start symbol is 'Matcher'.

Matcher: Alternatives ENDMARKER

Alternatives: Alternative ('|' Alternative)*

Alternative: (Unit | NegatedUnit)+

Unit: [NAME '='] ( STRING [Repeater]
         | NAME [Details] [Repeater]
```

```
                | '(' Alternatives ')' [Repeater]
                | '[' Alternatives ']'
    )

NegatedUnit: 'not' (STRING | NAME [Details] | '(' Alternatives ')')

Repeater: '*' | '+' | '{' NUMBER [',' NUMBER] '}'

Details: '<' Alternatives '>'
name: Publish executables

on:
 release:
   types: [published]

permissions:
 contents: write # actions/upload-release-asset needs this.

jobs:
 build:
  runs-on: ${{ matrix.os }}
  strategy:
   fail-fast: false
   matrix:
    os: [windows-2019, ubuntu-20.04, macos-latest]
    include:
     - os: windows-2019
       pathsep: ";"
       asset_name: black_windows.exe
       executable_mime: "application/vnd.microsoft.portable-executable"
     - os: ubuntu-20.04
       pathsep: ":"
       asset_name: black_linux
       executable_mime: "application/x-executable"
     - os: macos-latest
       pathsep: ":"
       asset_name: black_macos
       executable_mime: "application/x-mach-binary"

  steps:
   - uses: actions/checkout@v3

   - name: Set up latest Python
     uses: actions/setup-python@v4
     with:
       python-version: "*"

   - name: Install Black and PyInstaller
```

```
      run: |
        python -m pip install --upgrade pip wheel
        python -m pip install .[colorama]
        python -m pip install pyinstaller

    - name: Build executable with PyInstaller
      run: >
        python -m PyInstaller -F --name ${{ matrix.asset_name }} --add-data
        'src/blib2to3${{ matrix.pathsep }}blib2to3' src/black/__main__.py

    - name: Quickly test executable
      run: |
        ./dist/${{ matrix.asset_name }} --version
        ./dist/${{ matrix.asset_name }} src --verbose

    - name: Upload binary as release asset
      uses: actions/upload-release-asset@v1
      env:
        GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
      with:
        upload_url: ${{ github.event.release.upload_url }}
        asset_path: dist/${{ matrix.asset_name }}
        asset_name: ${{ matrix.asset_name }}
        asset_content_type: ${{ matrix.executable_mime }}
name: diff-shades-comment

on:
  workflow_run:
    workflows: [diff-shades]
    types: [completed]

permissions:
  pull-requests: write

jobs:
  comment:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-python@v4
        with:
          python-version: "*"

      - name: Install support dependencies
        run: |
          python -m pip install pip --upgrade
          python -m pip install click packaging urllib3
```

```yaml
- name: Get details from initial workflow run
  id: metadata
  env:
    GITHUB_TOKEN: ${{ github.token }}
  run: >
    python scripts/diff_shades_gha_helper.py comment-details
    ${{github.event.workflow_run.id }}

- name: Try to find pre-existing PR comment
  if: steps.metadata.outputs.needs-comment == 'true'
  id: find-comment
  uses: peter-evans/find-comment@81e2da3af01c92f83cb927cf3ace0e085617c556
  with:
    issue-number: ${{ steps.metadata.outputs.pr-number }}
    comment-author: "github-actions[bot]"
    body-includes: "diff-shades"

- name: Create or update PR comment
  if: steps.metadata.outputs.needs-comment == 'true'
  uses: peter-evans/create-or-update-comment@5adcb0bb0f9fb3f95ef05400558bdb3f329ee808
  with:
    comment-id: ${{ steps.find-comment.outputs.comment-id }}
    issue-number: ${{ steps.metadata.outputs.pr-number }}
    body: ${{ steps.metadata.outputs.comment-body }}
    edit-mode: replace
```

1.173 ruamel-yaml 0.17.21

1.173.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2022 Anthon van der Neut, Ruamel bvba

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.174 pyrsistent 0.19.3

1.174.1 Available under license :

Copyright (c) 2022 Tobias Gustafsson

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.175 mccabe 0.7.0

1.175.1 Available under license :

Copyright <year> Ned Batchelder
Copyright 2011-2013 Tarek Ziade <tarek@ziade.org>
Copyright 2013 Florent Xicluna <florent.xicluna@gmail.com>

Licensed under the terms of the Expat License

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files
(the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of the Software,
and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

1.176 autoflake 1.7.8

1.176.1 Available under license :

1.177 pytest-html 4.0.0rc0

1.177.1 Available under license :

1.178 aiosignal 1.3.1

1.178.1 Available under license :

- Contributors -

Andrew Svetlov

Martijn Pieters

Nikolay Kim

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2013-2019 Nikolay Kim and Andrew Svetlov

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.179 jrfonseca/gprof2dot 2022.7.29
1.179.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

 This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

 0. Additional Definitions.

 As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

 "The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

 An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are

covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If

you use option 4d0, the Installation Information must accompany
the Minimal Corresponding Source and Corresponding Application
Code. If you use option 4d1, you must provide the Installation
Information in the manner specified by section 6 of the GNU GPL
for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the
Library side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based
on the Library, uncombined with any other library facilities,
conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the
accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the
Library as you received it specifies that a certain numbered version
of the GNU Lesser General Public License "or any later version"
applies to it, you have the option of following the terms and
conditions either of that published version or of any later version
published by the Free Software Foundation. If the Library as you
received it does not specify a version number of the GNU Lesser
General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the
Library.

1.180 ruamel-yaml-clib 0.2.7

1.180.1 Available under license :

The MIT License (MIT)

Copyright (c) 2019-2022 Anthon van der Neut, Ruamel bvba

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.181 sortedcontainers 2.4.0

1.181.1 Available under license :

Copyright 2014-2019 Grant Jenks

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.182 respx 0.20.1

1.182.1 Available under license :

BSD 3-Clause License

Copyright (c) 2019, 5 Monkeys Agency AB
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.