

Flake8

Christian Külker

2020-05-08

Contents

1	Installation	2
2	Versions	2
3	Usage	2
3.1	Single File Usage	2
3.2	Multiple File Usage	2
3.3	Output	3
4	Useful Command Line Options	3
5	Ignoring Errors (Violations)	3
5.1	Ignoring Files Or Directory	3
5.2	Ignoring Error (Violation) Types	3
5.3	Ignoring Lines	3
6	Discussion	4
7	History	4
8	Disclaimer of Warranty	4
9	Limitation of Liability	4

With **flake8** it is possible enforcing a style guide to hand written python code. By default **flake8** includes checks provided by **PyFlakes**, PEP-0008 inspired style checks provided by **PyCodeStyle**, and **McCabe** complexity checking.

1 Installation

```
aptitude install flake8
```

2 Versions

Debian Stretch

```
flake8 --version
3.2.1 (mccabe: 0.5.3, pyflakes: 1.3.0, pycodestyle: 2.2.0) CPython 3.5.3
↪ on Linux
```

Debian Buster

```
flake8 --version
3.6.0 (mccabe: 0.6.1, pycodestyle: 2.4.0, pyflakes: 2.0.0) CPython 3.7.3
↪ on Linux
```

Debian Bullseye

```
flake8 --version
3.8.4 (mccabe: 0.6.1, pycodestyle: 2.6.0, pyflakes: 2.2.0) CPython 3.9.2
↪ on Linux
```

3 Usage

The command line tool `flake8` can be used on a single file or on multiple files.

3.1 Single File Usage

```
python3 flake8 /PATH/TO/file.py
```

3.2 Multiple File Usage

```
python3 flake8 /PATH/TO/FILES/*.py
```

3.3 Output

The command `flake8` prints a list of issues found. While most of them are related to style it also prints if it found a variable assigned but not used. (Perl is doing this automatically via `-w` option).

4 Useful Command Line Options

While **flake8** is usable already without options, it can be customized via a configuration file or various options. This sections shows only a small subset of useful command line options.

- `-q --quite` : Just print the name of file in which issues are found
- `--config=CONFIG` : specify the configuration file (for custom options)
- `--max-line-length=n` : set different line length (default 79)

5 Ignoring Errors (Violations)

5.1 Ignoring Files Or Directory

To ignore whole files or directories

```
flake8 --exclude path/to/file
```

5.2 Ignoring Error (Violation) Types

To ignore certain error (violation) types, it is possible to provide the ID on the command line. (It is also possible to it in a configuration file)

New exclude list:

```
flake8 --ignore=E1,E23,W503 path/to/files/
```

Extending the exclude list:

```
flake8 --extend-ignore=E1,E23 path/to/files/
```

5.3 Ignoring Lines

If the line is short enough it is possible to ignore one line with a special comment `# noqa`.

```
example = lambda: 'example' # noqa: E731
```

6 Discussion

Having **flake8** under Python is nice to have. Compared with other solutions, like **perltidy**, it seems this tool needs to go a long way before it is as easy to use, useful and time saving. A short example: When using an editor (like vim) it is possible to define a hot key to activate **perltidy** to format the displayed code. So as a programmer one do not need to write all the indentations and other fancy spaces or alignments. Just hit that key and the code looks beautiful. With **flake8** the programmer gets just a report and has to fix all manually. **perltidy** can batch convert files, check files embedded in other markup (HTML for example) and much more. As Python has in contrast to Perl the syntax philosophy “there is only one way” it is of course no wonder that **flake8** do not offer much to customize syntax, but one real advantage of **flake8** is, that it print out IDs referencing the rules, that can be used up to look up and find more information about specific rules.

7 History

Version	Date	Notes
0.1.1	2022-07-06	History, shell->bash, Debian 11 Bullseye
0.1.0	2020-05-08	Initial release

8 Disclaimer of Warranty

THERE IS NO WARRANTY FOR THIS INFORMATION, DOCUMENTS AND PROGRAMS, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE INFORMATION, DOCUMENT OR 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 INFORMATION, DOCUMENTS AND PROGRAMS IS WITH YOU. SHOULD THE INFORMATION, DOCUMENTS OR PROGRAMS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

9 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 INFORMATION, DOCUMENTS OR PROGRAMS 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 INFORMATION, DOCUMENTS OR PROGRAMS (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 INFORMATION, DOCUMENTS OR PROGRAMS TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.