
highlight

Release 0.2.0

Highlight text on the command line.

Dominic Davis-Foster

May 15, 2024

Contents

1	Installation	1
1.1	from GitHub	1
2	highlight	3
2.1	Examples	4
3	Contributing	5
3.1	Coding style	5
3.2	Automated tests	5
3.3	Type Annotations	5
3.4	Build documentation locally	6
4	Downloading source code	7
4.1	Building from source	8
5	License	9
	Index	11

Installation

1.1 from GitHub

```
$ python3 -m pip install git+https://github.com/domdfcoding/highlight@master --user
```


highlight

Print the content of FILE with syntax highlighting.

With no FILE, or when FILE is -, read standard input.

```
highlight [OPTIONS] [FILE]
```

Options

- m, --mime** <mime>
The MIME type to use for highlighting.
- l, --lexer** <lexer>
The Pygments lexer to use for highlighting.
- T, --show-tabs**
Display TAB characters as ^I.
- E, --show-ends**
Display \$ at the end of each line.
- n, --number**
Number all output lines.
- D, --debug**
Print the lexer name to stderr.
- version**
Show the version and exit.

Arguments

FILE
Optional argument. Default None

Notes

- `--lexer` and `--mime` are mutually exclusive.
- If no `--lexer` or `--mime` option is given, the lexer is determined from FILE. If reading from standard input, or if the file extension is unknown, the lexer is determined by examining the file's contents.
- If the `--mime` option is given but no lexer can be found for that mime type the mime type is ignored.
- See <https://pygments.org/docs/lexers/> for a list of available lexers.

2.1 Examples

Highlight a Python source code file.

```
highlight code.py
```

Highlight a Ruby source code file with the `RubyConsoleLexer`.

```
highlight code.rb --lexer rbcon
```

Highlight a Rust source code file with line numbers.

```
highlight main.rs --number
```


Contributing

highlight uses `tox` to automate testing and packaging, and `pre-commit` to maintain code quality.

Install `pre-commit` with `pip` and install the git hook:

```
$ python -m pip install pre-commit
$ pre-commit install
```

3.1 Coding style

`formate` is used for code formatting.

It can be run manually via `pre-commit`:

```
$ pre-commit run formate -a
```

Or, to run the complete autoformatting suite:

```
$ pre-commit run -a
```

3.2 Automated tests

Tests are run with `tox` and `pytest`. To run tests for a specific Python version, such as Python 3.6:

```
$ tox -e py36
```

To run tests for all Python versions, simply run:

```
$ tox
```

3.3 Type Annotations

Type annotations are checked using `mypy`. Run `mypy` using `tox`:

```
$ tox -e mypy
```

3.4 Build documentation locally

The documentation is powered by Sphinx. A local copy of the documentation can be built with `tox`:

```
$ tox -e docs
```

Downloading source code

The highlight source code is available on GitHub, and can be accessed from the following URL: <https://github.com/domdfcoding/highlight>

If you have git installed, you can clone the repository with the following command:

```
$ git clone https://github.com/domdfcoding/highlight
```

```
Cloning into 'highlight'...
remote: Enumerating objects: 47, done.
remote: Counting objects: 100% (47/47), done.
remote: Compressing objects: 100% (41/41), done.
remote: Total 173 (delta 16), reused 17 (delta 6), pack-reused 126
Receiving objects: 100% (173/173), 126.56 KiB | 678.00 KiB/s, done.
Resolving deltas: 100% (66/66), done.
```

Alternatively, the code can be downloaded in a 'zip' file by clicking:

Clone or download → Download Zip

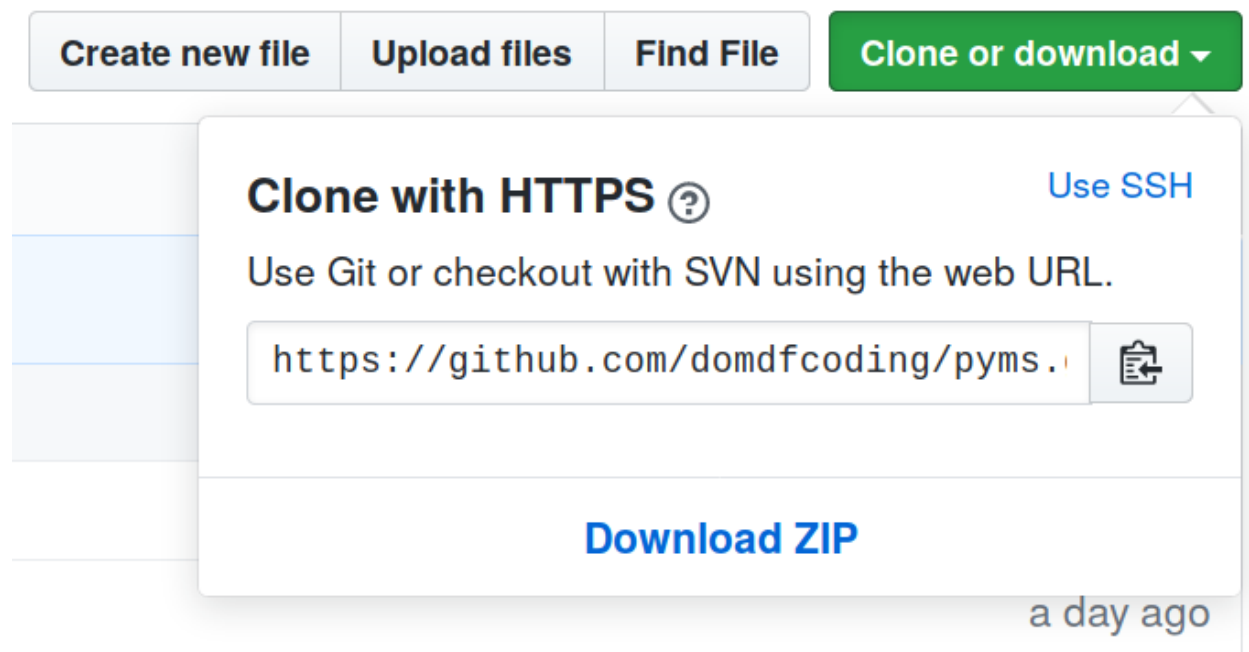


Fig. 1: Downloading a 'zip' file of the source code

4.1 Building from source

The recommended way to build `highlight` is to use `tox`:

```
$ tox -e build
```

The source and wheel distributions will be in the directory `dist`.

If you wish, you may also use `pep517.build` or another **PEP 517**-compatible build tool.

License

highlight is licensed under the [MIT License](#)

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- Commercial use – The licensed material and derivatives may be used for commercial purposes.
- Modification – The licensed material may be modified.
- Distribution – The licensed material may be distributed.
- Private use – The licensed material may be used and modified in private.

Conditions

- License and copyright notice – A copy of the license and copyright notice must be included with the licensed material.

Limitations

- Liability – This license includes a limitation of liability.
- Warranty – This license explicitly states that it does NOT provide any warranty.

[See more information on choosealicense.com](#) ⇒

```
Copyright (c) 2021 Dominic Davis-Foster
```

```
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.
```


Symbols

-D highlight command line option, 3
-E highlight command line option, 3
-T highlight command line option, 3
--debug highlight command line option, 3
--lexer <lexer> highlight command line option, 3
--mime <mime> highlight command line option, 3
--number highlight command line option, 3
--show-ends highlight command line option, 3
--show-tabs highlight command line option, 3
--version highlight command line option, 3
-l highlight command line option, 3
-m highlight command line option, 3
-n highlight command line option, 3

F

FILE highlight command line option, 3

H

highlight command line option

-D, 3
-E, 3
-T, 3
--debug, 3
--lexer <lexer>, 3
--mime <mime>, 3
--number, 3
--show-ends, 3
--show-tabs, 3
--version, 3

-l, 3
-m, 3
-n, 3
FILE, 3

M

MIT License, 9

P

Python Enhancement Proposals
PEP 517, 8