
Epithet Documentation

Release 0.0.3

Philip James

Jun 13, 2017

Contents

1	Epithet	3
1.1	Features	3
1.2	Credits	3
2	Installation	5
2.1	Stable release	5
2.2	From sources	5
3	Usage	7
4	Contributing	9
4.1	Types of Contributions	9
4.2	Get Started!	10
4.3	Pull Request Guidelines	11
4.4	Tips	11
5	Credits	13
5.1	Development Lead	13
5.2	Contributors	13
6	History	15
6.1	0.0.3 (2017-06-11)	15
6.2	0.0.2 (2017-05-25)	15
6.3	0.0.1 (2017-05-25)	15
7	Indices and tables	17

Contents:

CHAPTER 1

Epithet

Manage your labels.

- Free software: MIT license
- Documentation: <https://epithet.readthedocs.io>.

Features

- Allows you to list all the labels on a repo, or all repos on an organization.

Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.

Stable release

To install Epithet, run this command in your terminal:

```
$ pip install epithet
```

This is the preferred method to install Epithet, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

From sources

The sources for Epithet can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/phildini/epithet
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/phildini/epithet/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```


CHAPTER 3

Usage

These docs are a little lacking at the moment, and we'd love to see them improved!

You can use the epithet commands without a key, but you might quickly run into rate limiting issues. Get a personal token from [Github](#) and pass it into the `--key` option to solve this.

To use Epithet in a project:

```
$ epithet

Usage: epithet [OPTIONS] COMMAND [ARGS]...

Options:
  --dryrun  Don't actually change or create labels
  --help    Show this message and exit.

Commands:
  add
  list

$ epithet add --help
Usage: epithet add [OPTIONS]

Options:
  --name TEXT  Name of new label
  --color TEXT Color of new label
  --key TEXT   OAuth Token
  --org TEXT   Organization to get repos from
  --repo TEXT  Optionally select a single repo
  --help      Show this message and exit.
```


Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

Types of Contributions

Report Bugs

Report bugs at <https://github.com/phildini/epithet/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

Write Documentation

Epithet could always use more documentation, whether as part of the official Epithet docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/phildini/epithet/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

Get Started!

Ready to contribute? Here's how to set up *epithet* for local development.

1. Fork the *epithet* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/epithet.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv epithet
$ cd epithet/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 epithet tests
$ python setup.py test or py.test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/phildini/epithet/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

```
$ python -m unittest tests.test_epithet
```


CHAPTER 5

Credits

Development Lead

- Philip James <phildini@phildini.net>

Contributors

None yet. Why not be the first?

0.0.3 (2017-06-11)

- Add “delete” command
- Allow specifying an API URL for GitHub Enterprise
- Make *-key* param required

0.0.2 (2017-05-25)

- Improved rate limit handling

0.0.1 (2017-05-25)

- First release on PyPI.

CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`