
Flask-Deprecate Documentation

Release 0.1.3

Tim Martin

Feb 08, 2018

Contents

1	Installation	3
1.1	Stable release	3
1.2	From sources	3
2	Usage	5
3	Contributing	7
3.1	Types of Contributions	7
3.2	Get Started!	8
3.3	Pull Request Guidelines	9
3.4	Tips	9
4	History	11
4.1	0.1.0 (2018-02-07)	11
5	flask_deprecate API	13
5.1	Submodules	13
5.2	flask_deprecate.flask_deprecate module	13
5.3	Module contents	14
6	Flask-Deprecate	17
6.1	Example	17
6.2	Documentation	18
6.3	Installation	18
6.4	Credits	18
7	Indices and tables	19
	Python Module Index	21

Contents:

1.1 Stable release

To install Flask-Deprecate, run this command in your terminal:

```
$ pip install flask_deprecate
```

This is the preferred method to install Flask-Deprecate, 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.

1.2 From sources

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

You can either clone the public repository:

```
$ git clone git://github.com/timmartin19/flask_deprecate
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/timmartin19/flask_deprecate/tarball/master
```

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

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


CHAPTER 2

Usage

To use Flask-Deprecate in a project:

```
import flask_deprecate
```


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

3.1 Types of Contributions

3.1.1 Report Bugs

Report bugs at <https://github.com/timmartin19/flask-deprecate/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.

3.1.2 Fix Bugs

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

3.1.3 Implement Features

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

3.1.4 Write Documentation

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

3.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/timmartin19/flask-deprecate/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 :)

3.2 Get Started!

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

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

```
$ git clone git@github.com:your_name_here/flask_deprecate.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 flask_deprecate
$ cd flask_deprecate/
$ 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 flask_deprecate 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.

3.3 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/timmartin19/flask-deprecate/pull_requests and make sure that the tests pass for all supported Python versions.

3.4 Tips

To run a subset of tests:

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


4.1 0.1.0 (2018-02-07)

- First release on PyPI.

5.1 Submodules

5.2 flask_deprecate.flask_deprecate module

Business logic. Kept separate in case the project becomes larger for some reason in the future.

```
flask_deprecate.flask_deprecate.deprecate_blueprint(old_blueprint,  
                                                    new_blueprint=None,    mes-  
                                                    sage=")
```

Deprecates an every route on a blueprint by adding a header “WARNING” with a message.

If a `url_prefix` is set for the blueprint in either the blueprint init or the registration with the app, then an additional section on the header will appear saying the blueprint is deprecated. If a `new_blueprint` is also provided, the header will have additional info directing the client to the new URL. Note, that this will not automatically redirect the client.

The blueprint must be deprecated before registering it with the application

```
>>> from flask import Flask, Blueprint
>>> app = Flask('myapp')
>>> deprecated_bp = Blueprint('deprecated', 'deprecated')
>>> new_bp = Blueprint('new', 'new')
>>> deprecate_blueprint(old_blueprint, new_blueprint=new_bp)
>>> app.register_blueprint(bp, url_prefix='/v1')
>>> app.register_blueprint(new_bp, url_prefix='/v2')
```

Parameters

- **old_blueprint** (*flask.Blueprint*) – The blueprint to be deprecated
- **new_blueprint** (*flask.Blueprint*) – The new blueprint that will be replacing the old one (if applicable).
- **message** (*str*) – An extra message to append to the warning header

Return type NoneType

`flask_deprecate.flask_deprecate.deprecate_route(message=)`

Adds a Warning header to the response informing the client that the route is deprecated

```
from flask import Flask, Response
from flask_deprecate import deprecate_route

app = Flask('myapp')

@app.route('/my_route')
@deprecate_route()
def deprecated_route():
    return Response()
```

Parameters `message (str)` – Additional information to provide to the client

5.3 Module contents

A tool for deprecating APIs in Flask. Injects a head for clients to catch and indicates the upgrade path.

`flask_deprecate.deprecate_blueprint(old_blueprint, new_blueprint=None, message=)`

Deprecates an every route on a blueprint by adding a header “WARNING” with a message.

If a `url_prefix` is set for the blueprint in either the blueprint init or the registration with the app, then an additional section on the header will appear saying the blueprint is deprecated. If a `new_blueprint` is also provided, the header will have additional info directing the client to the new URL. Note, that this will not automatically redirect the client.

The blueprint must be deprecated before registering it with the application

```
>>> from flask import Flask, Blueprint
>>> app = Flask('myapp')
>>> deprecated_bp = Blueprint('deprecated', 'deprecated')
>>> new_bp = Blueprint('new', 'new')
>>> deprecate_blueprint(old_blueprint, new_blueprint=new_bp)
>>> app.register_blueprint(bp, url_prefix='/v1')
>>> app.register_blueprint(new_bp, url_prefix='/v2')
```

Parameters

- **old_blueprint** (`flask.Blueprint`) – The blueprint to be deprecated
- **new_blueprint** (`flask.Blueprint`) – The new blueprint that will be replacing the old one (if applicable).
- **message** (`str`) – An extra message to append to the warning header

Return type NoneType

`flask_deprecate.deprecate_route(message=)`

Adds a Warning header to the response informing the client that the route is deprecated

```
from flask import Flask, Response
from flask_deprecate import deprecate_route

app = Flask('myapp')
```

```
@app.route('/my_route')
@deprecate_route()
def deprecated_route():
    return Response()
```

Parameters **message** (*str*) – Additional information to provide to the client

Easy decorators for deprecating flask views and blueprints

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

```
pip install flask-deprecate
```

6.1 Example

```
from flask import Flask, Response

from flask_deprecate import deprecate_view

app = Flask('myapp')

@app.route('/myroute')
@deprecate_view("Don't use this!")
def myroute():
    return Response()
```

An HTTP compliant “Warning” header is injected indicating the route is deprecated and optionally providing an upgrade path.

You can also deprecate an entire blueprint in favor of a new one

```
from flask import Flask, Response, Blueprint

from flask_deprecate import deprecate_blueprint

old_bp = Blueprint('old', 'old', url_prefix='/v1')
new_bp = Blueprint('new', 'new', url_prefix='/v2')
```

```
@old_bp.route('/my_route')
def my_old_route():
    return Resonse()

@new_bp.route('/my_new_route')
def my_new_route():
    return Response()

deprecate_blueprint(old_bp, new_blueprint=new_bp)
app.register_blueprint(old_bp)
app.register_blueprint(new_bp)
```

This will inject the Warning header for every route on the old blueprint and additionally direct the client to use the new /v2 api.

6.2 Documentation

You will need to install the package dependencies first, see the Installation section for details.

To build and open the documentation simply run:

```
bin/build-docs
```

6.3 Installation

If you need to install pyenv/virtualenvwrapper you can run the *bin/setup-osx* command Please note that this will modify your bash profile

Assuming you have virtualenv wrapper installed

```
mkvirtualenv flask-deprecate
workon flask-deprecate
pip install -r requirements_dev.txt
pip install -e .
```

6.4 Credits

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

CHAPTER 7

Indices and tables

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

f

`flask_deprecate`, [14](#)

`flask_deprecate.flask_deprecate`, [13](#)

D

`deprecate_blueprint()` (in module `flask_deprecate`), [14](#)
`deprecate_blueprint()` (in module `flask_deprecate.flask_deprecate`), [13](#)
`deprecate_route()` (in module `flask_deprecate`), [14](#)
`deprecate_route()` (in module `flask_deprecate.flask_deprecate`), [14](#)

F

`flask_deprecate` (module), [14](#)
`flask_deprecate.flask_deprecate` (module), [13](#)