envcat Documentation

Release 0.1.1

Paul McLanahan

May 12, 2017

Contents

Contents:

CHAPTER 1

envcat

Mix and merge env file values.

- Free software: Apache Software License 2.0
- Documentation: https://envcat.readthedocs.io.

Features

- Merge keys and values from any number of env files.
- Ignore missing env files.
- Output to STDOUT in env file format or a single line (good for passing to commands like *heroku config*).

Credits

This package was created with Cookiecutter and the audreyr/cookiecutter-pypackage project template.

CHAPTER 2

Installation

Stable release

To install envcat, run this command in your terminal:

```
$ pip install envcat
```

This is the preferred method to install envcat, 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 envcat can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/pmac/envcat
```

Or download the tarball:

```
$ curl  -OL https://github.com/pmac/envcat/tarball/master
```

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

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

CHAPTER 3

Usage

envcat is a CLI tool for mixing the content of one or more env files:

```
$ envcat base.env local.env > mix.env
```

In this example values would be read from *base.env* and *local.env* in that order. Order is important since the same key in *local.env* would override one from *base.env* in this order. If any of the specified files do not exist they are silently ignored, making this a good tool for searching an optionally including values from files that may exist.

The default output format is that of an env file (like foreman uses), but you can also have it output the values space separated on a single line using the *–oneline* flag:

```
$ envcat --help
usage: envcat.py [-h] [--oneline] [FILE [FILE ...]]

Merge env file values

positional arguments:
  FILE        env file paths

optional arguments:
  -h, --help  show this help message and exit
  --oneline   Output the env variables on one line
  --version   Output the version and exit
```

This tool can be particularly useful when passing values to other CLI tools:

```
#!/bin/bash

ENV_VALUES=( $(envcat base.env ${REGION}.env local.env) )
# add a value from the script
ENV_VALUES+=( "OTHER_VALUE=$OTHER_ENV_VAR" )
heroku config:set -a my-app "${ENV_VALUES[@]}"
```

Docker

You can also easily use this tool via docker if you don't want to install it locally. Just add the following to your project as a script called *envcat* and it'll work just like the tool itself:

```
#!/bin/bash

docker run --rm -v $(pwd):/envcat mozmeao/envcat:latest "$@"
```

Then you can use it like normal:

```
$ ./envcat --version
0.1.0
```

CHAPTER 4

Contributing

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/pmac/envcat/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

envcat could always use more documentation, whether as part of the official envcat 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/pmac/envcat/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 *envcat* for local development.

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

   ```
   $ git clone git@github.com:your_name_here/envcat.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 envcat
   $ cd envcat/
   $ 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 envcat.py tests
   $ 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/pmac/envcat/pull_requests and make sure that the tests pass for all supported Python versions.

CHAPTER 5

Credits

Development Lead

- Paul McLanahan <pmac@mozilla.com>

Contributors

None yet. Why not be the first?

CHAPTER 6

History

0.1.1 (2017-05-11)

- Add a dockerized version
- Add a –version argument

0.1.0 (2017-05-09)

- First release on PyPI.

CHAPTER 7

Indices and tables

- genindex
- modindex
- search