webchk Documentation

Release 1.0.4

Amged Rustom

Sep 05, 2020

Contents

Contents:

CHAPTER 1

webchk

webchk is a command-line tool developed in Python 3 for checking the HTTP status codes and response headers of URLs. It accepts one or more URLs as arguments. Furthermore, a sitemap URL can be passed using the -p option to download its content, extract the URLs and check their statuses.

1.1 Installation

webchk is available on PyPI and can be installed using pip with the following command:

```
$ pip install webchk
```

Webchk does not require any 3rd party packages to run. So it can also be cloned from GitHub and run as a module:

```
$ git clone https://github.com/amgedr/webchk.git
$ cd webchk
$ python3 -m webchk
```

1.2 Usage

```
webchk [-h] [-i INPUT] [-o OUTPUT] [-p] [-a] [-l] [-s] [-f] [-v]
              [urls [urls ...]]

positional arguments:
  urls
```

(continues on next page)

(continued from previous page)

```
optional arguments:
  -h, --help                  show this help message and exit
  -i INPUT, --input INPUT     Read input from a file
  -o OUTPUT, --output OUTPUT  Save output to a file
  -p, --parse                 Follow links listed in .xml URLs
  -l, --list                  Print URLs without checking them
  -v, --version               Print the version number
```

1.2.1 Examples

Check a list of URLs from a file (one URL per line):

```
$ webchk -i urls.txt
```

Check the status of a sitemap file and all the URLs listed in it:

```
$ webchk -p http://example.com/sitemap.xml
```

List the URLs in a file without checking their HTTP status:

```
$ webchk -li urls.txt
```

Check the URLs in a file and .xml files in it:

```
$ webchk -pi urls.txt
```

List the URLs in a file and .xml files in it:

```
$ webchk -pli urls.txt
```

List the URLs in a sitemap without checking their status:

```
$ webchk -lp http://example.com/sitemap.xml
```

CHAPTER 2

Installation

2.1 Stable release

To install webchk, run this command in your terminal:

```
$ pip install webchk
```

This is the preferred method to install webchk, 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.

2.2 From sources

The sources for webchk can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/amgedr/webchk
```

Or download the tarball:

```
$ curl  -OL https://github.com/amgedr/webchk/tarball/master
```

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

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

CHAPTER 3

Usage

To use webchk in a project:

```
import webchk
```

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:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/amgedr/webchk/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.

4.1.2 Fix Bugs

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

4.1.3 Implement Features

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

4.1.4 Write Documentation

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

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/amgedr/webchk/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 :)

4.2 Get Started!

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

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

```
$ git clone git@github.com:your_name_here/webchk.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 webchk
$ cd webchk/
$ pip install requirements_dev.txt
$ 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 webchk 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.

4.3 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 3.4, 3.5 and 3.6, and for PyPy. Check https://travis-ci.org/amgedr/webchk/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest test
```

CHAPTER 5

Credits

5.1 Development Lead

- Amged Rustom https://github.com/amgedr

5.2 Contributors

None yet. Why not be the first?

CHAPTER 6

History

6.1 1.0.4 (2020-08-29)

- Fix crash when an URL check times out
- Fix –output command-line option

6.2 1.0.3 (2020-07-10)

- Modify the Python versions tox tests
- Fix failing unit tests

6.3 1.0.2 (2020-05-08)

- Add Python versions 3.7 and 3.8 to the list of tested versions

6.4 1.0.1 (2020-05-08)

- Fixed: Parsing sitemaps enters an endless loop
- Fixed: Parsing a URL that does not exists exits with an unhandled exception

6.5 1.0.0 (2018-12-06)

- Linked to http://codehill.com/projects/webchk/ instead of readthedocs.io

6.6 0.3.0 (2018-03-24)

- Run each check in its own thread

6.7 0.2.1 (2017-12-19)

- Fixed: Status code description not being displayed
- Improved PyPI and GitHub README

6.8 0.2.0 (2017-12-14)

- Code refactoring
- Created setup.py

CHAPTER 7

Indices and tables

- genindex
- modindex
- search