myratings Documentation

Release 0.1.0

Arun KG

Nov 15, 2018

Contents

Contents:

myratings

Displays ratings of the user from various competitive programming sites in the terminal !!

- Free software: MIT license
- Documentation: https://myratings.readthedocs.io.

1.1 Features

- **Mode 1:** In this mode user handles and ratings are stored in the disk when user enters them for the first time so that the user need not enter them in the future and also it provides a comparison summary between old and new ratings.It is the default mode.
- **Mode 2:** If it is not possible to maintain log, users can still view their ratings by simply selecting competitive programming site option number and entering handle.

1.2 Installation

Type the following command in the terminal:

```
pip install myratings
```

1.3 Usage

- Type "myratings" from the commandline

For the first time users are supposed to enter their respective handles,

```
(venv) kg@arun-Aspire-V5-121:~$ myratings
Do you want to add hackerearth?? (y/n) y
Enter your handle: kgarun50
Do you want to add codechef?? (y/n) y
Enter your handle: kgarun
Do you want to add codeforces?? (y/n) y
Enter your handle: kgarun50
+---------------+------------+------------+----------+
|     Site      | OldRating  | NewRating  | Changes  |
+===============+============+============+==========+
| hackerearth   | NA         | 1549       | NA       |
+---------------+------------+------------+----------+
| codechef      | NA         | 1822       | NA       |
+---------------+------------+------------+----------+
| codeforces    | NA         | 1454       | NA       |
+---------------+------------+------------+----------+
```

On subsequent usage output will resemble the following :

```
(venv) kg@arun-Aspire-V5-121:~$ myratings
+---------------+------------+------------+----------+
|     Site      | OldRating  | NewRating  | Changes  |
+===============+============+============+==========+
| hackerearth   | 1549       | 1549       | 0        |
+---------------+------------+------------+----------+
| codechef      | 1702       | 1822       | 120      |
+---------------+------------+------------+----------+
| codeforces    | 1454       | 1454       | 0        |
+---------------+------------+------------+----------+
```

1.4 Configuration

To add or update handle type the following command:

```
myratings config
```

1.5 To use myratings in a project :

Simply type the following import statement:

```
import myratings
```

1.6 Credits

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

Installation

2.1 Stable release

To install myratings, run this command in your terminal:

```
$ pip install myratings
```

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

You can either clone the public repository:

```
$ git clone git://github.com/kgarun/myratings
```

Or download the tarball:

```
$ curl  -OL https://github.com/kgarun/myratings/tarball/master
```

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

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

Usage

Installation:

```
pip install myratings
```

To get ratings

```
* Type "myratings" from the commandline
* First time it asks for user handle of the respective sites
* Then rating summary will be displayed
```

To use myratings in a project:

```
import myratings
```

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/kgarun/myratings/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

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

1. Fork the *myratings* repo on GitHub.

2. Clone your fork locally:

   ```
   $ git clone git@github.com:your_name_here/myratings.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 myratings
   $ cd myratings/
   $ 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 myratings 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 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/kgarun/myratings/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 tests.test_myratings
```

Credits

5.1 Development Lead

- Arun KG <kgarun50@gmail.com>

5.2 Contributors

None yet. Why not be the first?

History

6.1 0.1.0 (2017-11-29)

- First release on PyPI.

CHAPTER 7

Indices and tables

- genindex
- modindex
- search