

---

# ura Documentation

*Release 0.1.0*

**revesansparole**

February 04, 2016



<b>1</b>	<b>Install</b>	<b>3</b>
<b>2</b>	<b>Use</b>	<b>5</b>
<b>3</b>	<b>Contribute</b>	<b>7</b>
<b>4</b>	<b>Acknowledgments</b>	<b>9</b>
<b>5</b>	<b>Installation</b>	<b>11</b>
<b>6</b>	<b>Usage</b>	<b>13</b>
<b>7</b>	<b>Contributing</b>	<b>15</b>
7.1	Types of Contributions . . . . .	15
7.2	Get Started! . . . . .	16
7.3	Pull Request Guidelines . . . . .	16
7.4	Tips . . . . .	17
<b>8</b>	<b>Credits</b>	<b>19</b>
8.1	Development Lead . . . . .	19
8.2	Contributors . . . . .	19
<b>9</b>	<b>History</b>	<b>21</b>
<b>10</b>	<b>creation (doc.creation_date)</b>	<b>23</b>
<b>11</b>	<b>Indices and tables</b>	<b>25</b>



Contents:



---

## Install

---

Download sources and use setup:

```
$ python setup.py install  
or  
$ python setup.py develop
```





---

### Use

---

Simple usage:

```
from ura import *
```



---

**Contribute**

---

Fork this project on [github](#)



---

## Acknowledgments

---



---

## Installation

---

Download source then, at the command line:

```
$ python setup.py
```





---

**Usage**

---



---

## 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:

### 7.1 Types of Contributions

#### 7.1.1 Report Bugs

Report bugs at 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.

#### 7.1.2 Fix Bugs

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

#### 7.1.3 Implement Features

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

#### 7.1.4 Write Documentation

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

#### 7.1.5 Submit Feedback

The best way to send feedback is to file an issue at 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 :)

## 7.2 Get Started!

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

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

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

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

```
$ virtualenv dvlpt
$ dvlpt/script/activate
(dvlpt)$ python setup.py develop
```

4. Create a branch for local development (wip stands for work in progress):

```
(dvlpt)$ git checkout -b wip_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*:

```
(dvlpt)$ cd ura
(dvlpt) ura$ flake8
(dvlpt) ura$ nosetests
(dvlpt) ura$ 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 wip_name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

## 7.3 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 2.7, 3.4, and 3.5. Check [https://travis-ci.org/github.owner/ura/pull\\_requests](https://travis-ci.org/github.owner/ura/pull_requests) and make sure that the tests pass for all supported Python versions.

## 7.4 Tips

To run a subset of tests:

```
$ nosetests test/test_XXX
```



---

## Credits

---

### 8.1 Development Lead

- `base.author_name <base.author_email>`

### 8.2 Contributors

None yet. Why not be the first?





---

## History

---



---

**creation (doc.creation\_date)**

---

- First release on PyPI.



---

## Indices and tables

---

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