

# Structuring Your Python Project

...

Matt Bachmann

# Agenda

- Directory Structure
- Isolating Your Environment
- Installing Dependencies
- Packaging Your Project
- Releasing to the World

# Our Project

- Display Current Weather

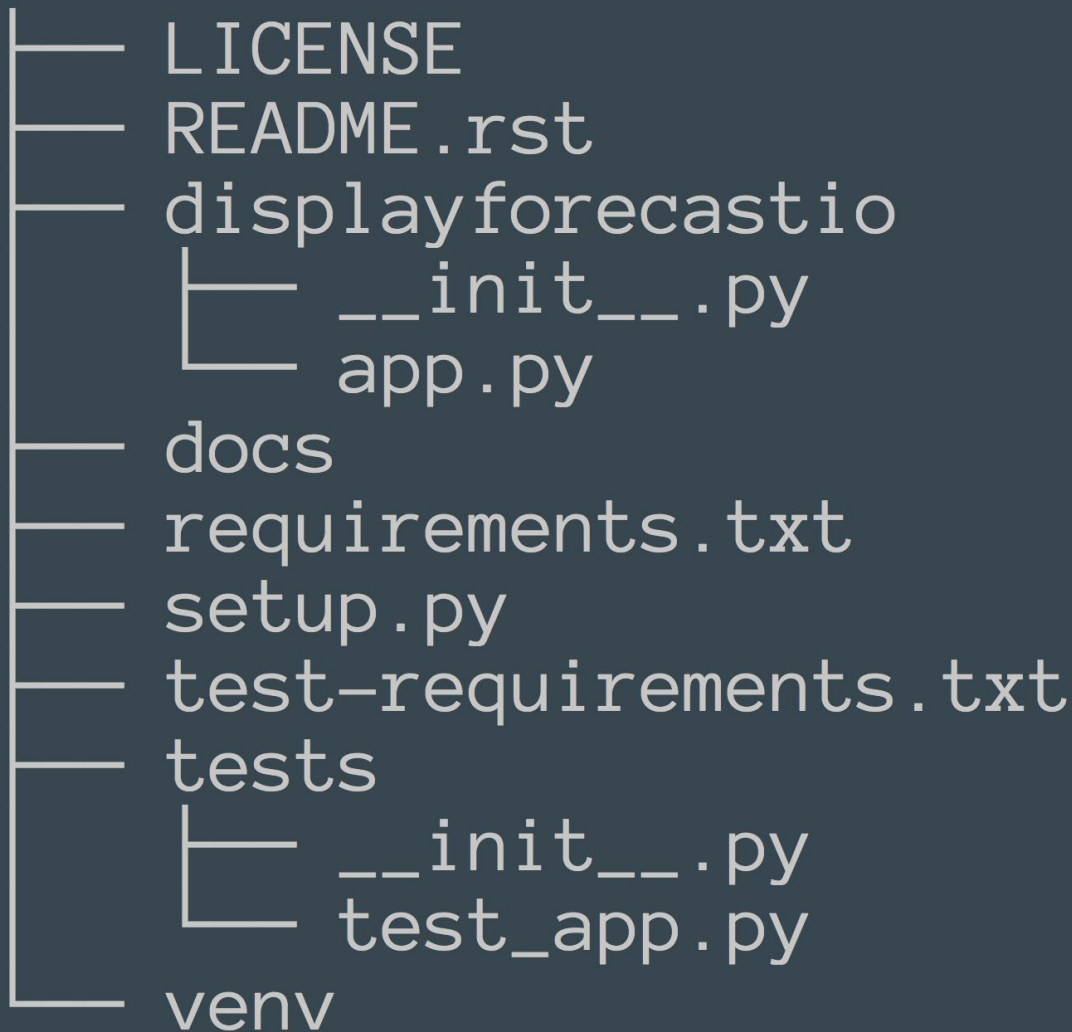
```
> forcastio 42.3907 -71.1157
```

Currently: rain - Mixed

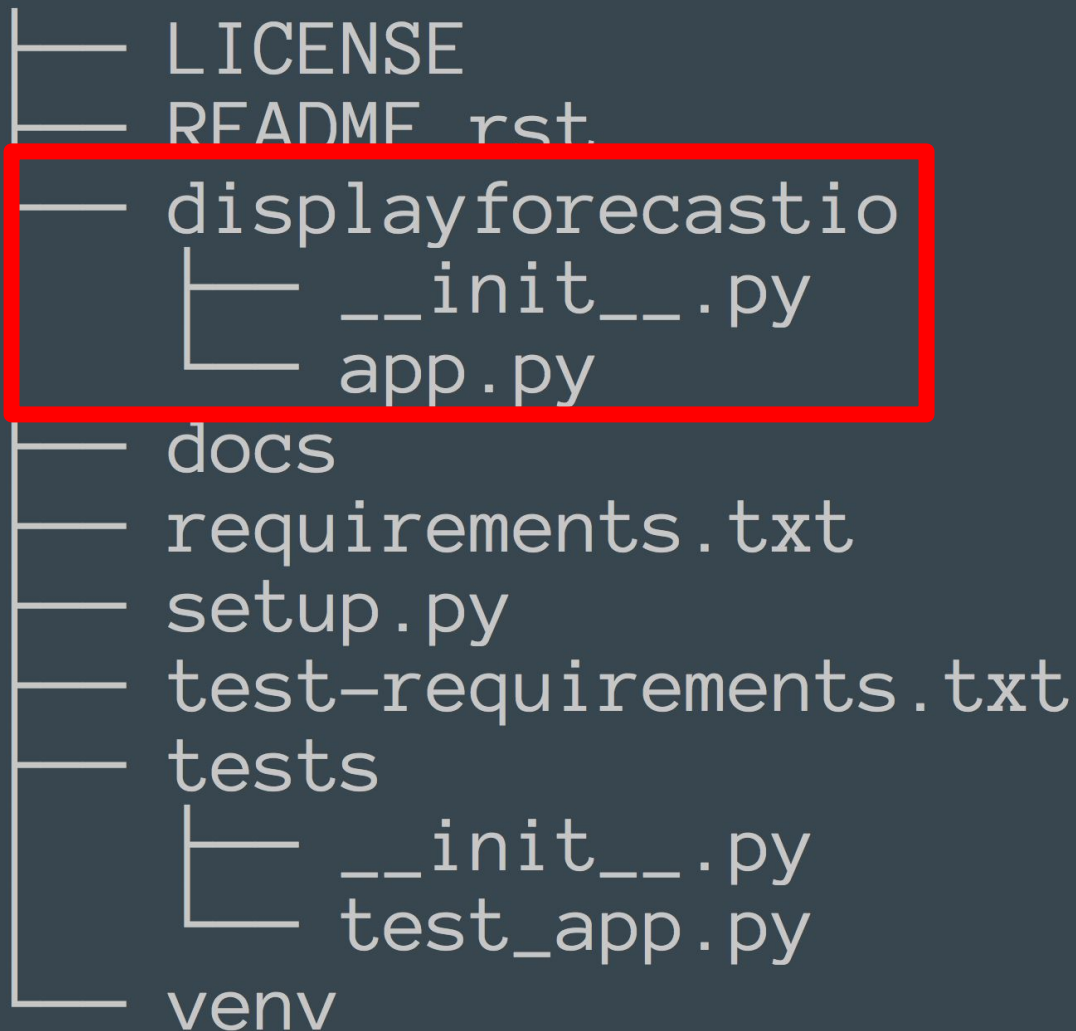
precipitation tomorrow through

Thursday, with temperatures falling  
to 41°F on Monday.

# Project Structure



# The Code



# Style Checking Your Code

```
> flake8 displayforecastio
```

```
displayforecastio/app.py:47:80: E501 line too long (81 > 79 characters)
```

# The Tests

```
graph LR; LICENSE --- Root; README_rst[README.rst] --- Root; displayforecastio --- Root; Root --- sub1[ ]; sub1 --- __init__1[__init__.py]; sub1 --- app_py[app.py]; docs --- Root; requirements_txt[requirements.txt] --- Root; setup_py[setup.py] --- Root; test_requirements_txt[test-requirements.txt] --- Root; Root --- tests; tests --- sub2[ ]; sub2 --- __init__2[__init__.py]; sub2 --- test_app_py[test_app.py]; venv --- Root
```

LICENSE

README.rst

displayforecastio

├── \_\_init\_\_.py

└── app.py

docs

requirements.txt

setup.py

test-requirements.txt

tests

├── \_\_init\_\_.py

└── test\_app.py

venv

# Running Your Tests

```
> py.test tests
```

```
===== test session starts =====
```

```
platform darwin -- Python 2.7.10, pytest-2.8.3, py-1.4.30, pluggy-0.3.1
```

```
rootdir: /Users/bachmann/code/terminalweather, inifile:
```

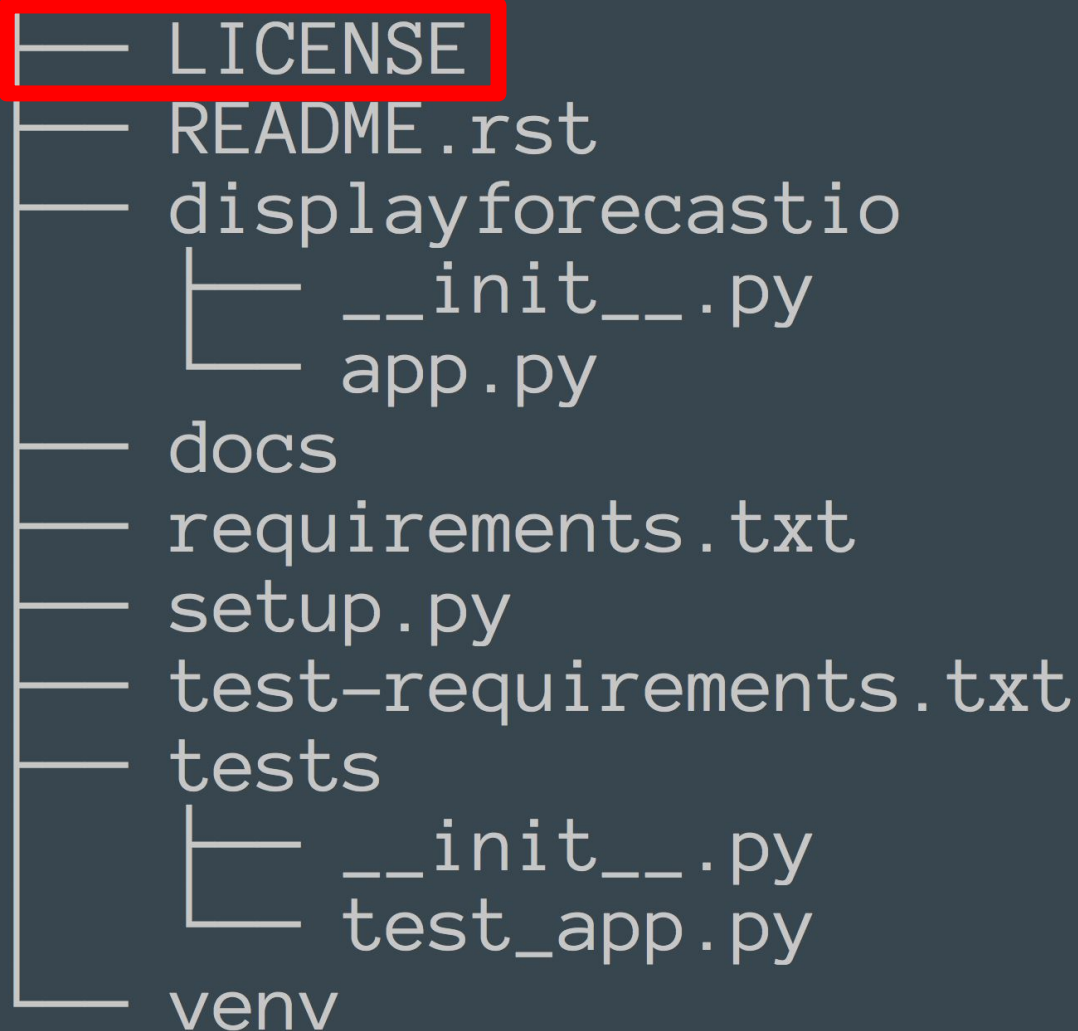
```
collected 4 items
```

```
tests/test_app.py ....
```

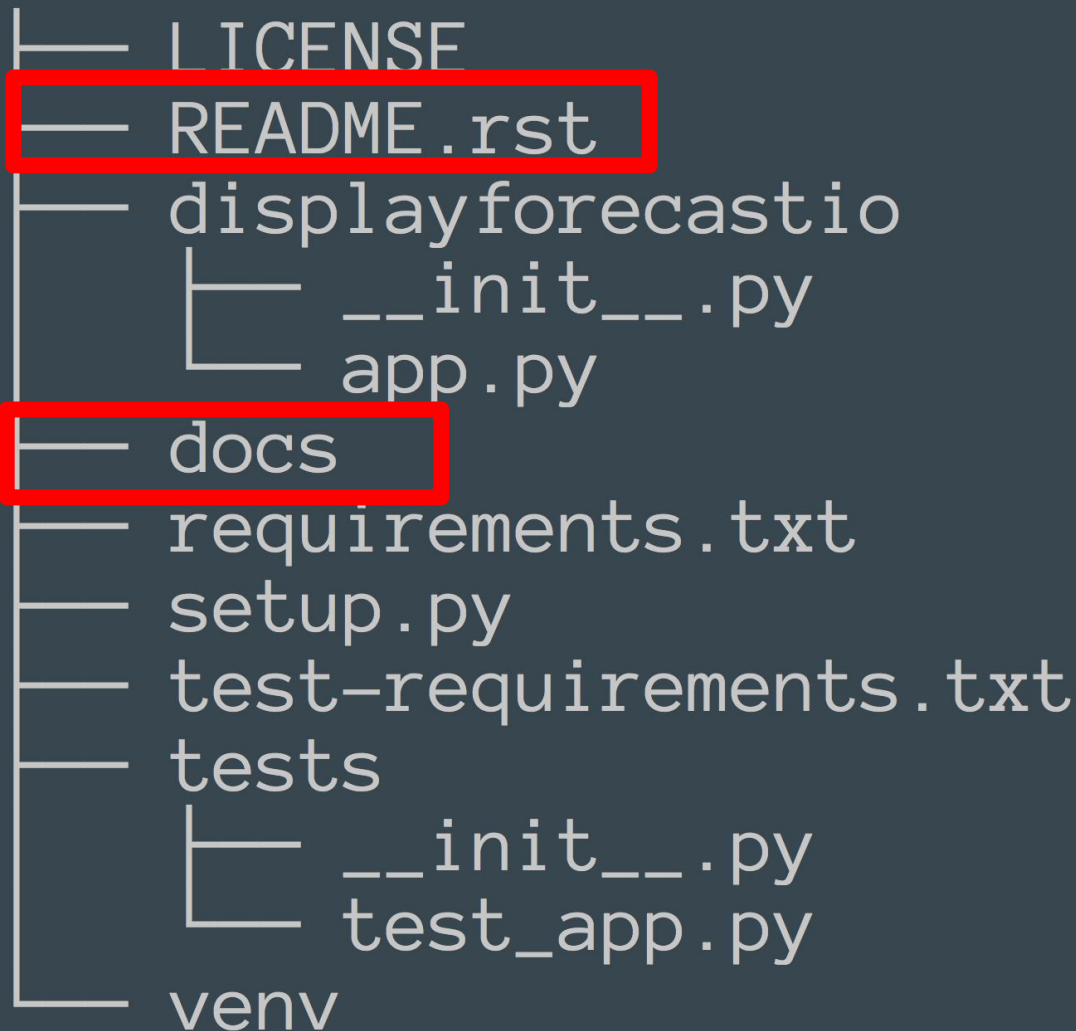
```
===== 4 passed in 1.59 seconds =====
```



# License



# Documentation



# Documentation

- All projects should have a readme
  - Description
  - Installation
  - Usage
- Moving Beyond Readme!
  - Sphinx
- Write docs in reStructuredText
  - Renders in Github and PyPi. PyPi does not support Markdown

# Specifying Dependencies and Isolating Your Project

```
graph TD; LICENSE --- Root; README_rst[README.rst] --- Root; displayforecastio --- Root; Root --- subgraph sub1 [ ]; __init__py1[__init__.py]; apppy[app.py]; end; docs --- Root; requirements_txt[requirements.txt]; setuppy[setup.py]; test_requirements_txt[test-requirements.txt]; tests --- Root; Root --- subgraph sub2 [ ]; __init__py2[__init__.py]; test_apppy[test_app.py]; end; venv --- Root;
```

LICENSE

README.rst

displayforecastio

- \_\_init\_\_.py
- app.py

docs

requirements.txt

setup.py

test-requirements.txt

tests

- \_\_init\_\_.py
- test\_app.py

venv

# Isolate Your Project

- Install virtualenv
  - > `pip install virtualenv`
- Create an isolated environment
  - > `virtualenv venv`
- Turn that environment on
  - > `source venv/bin/activate`

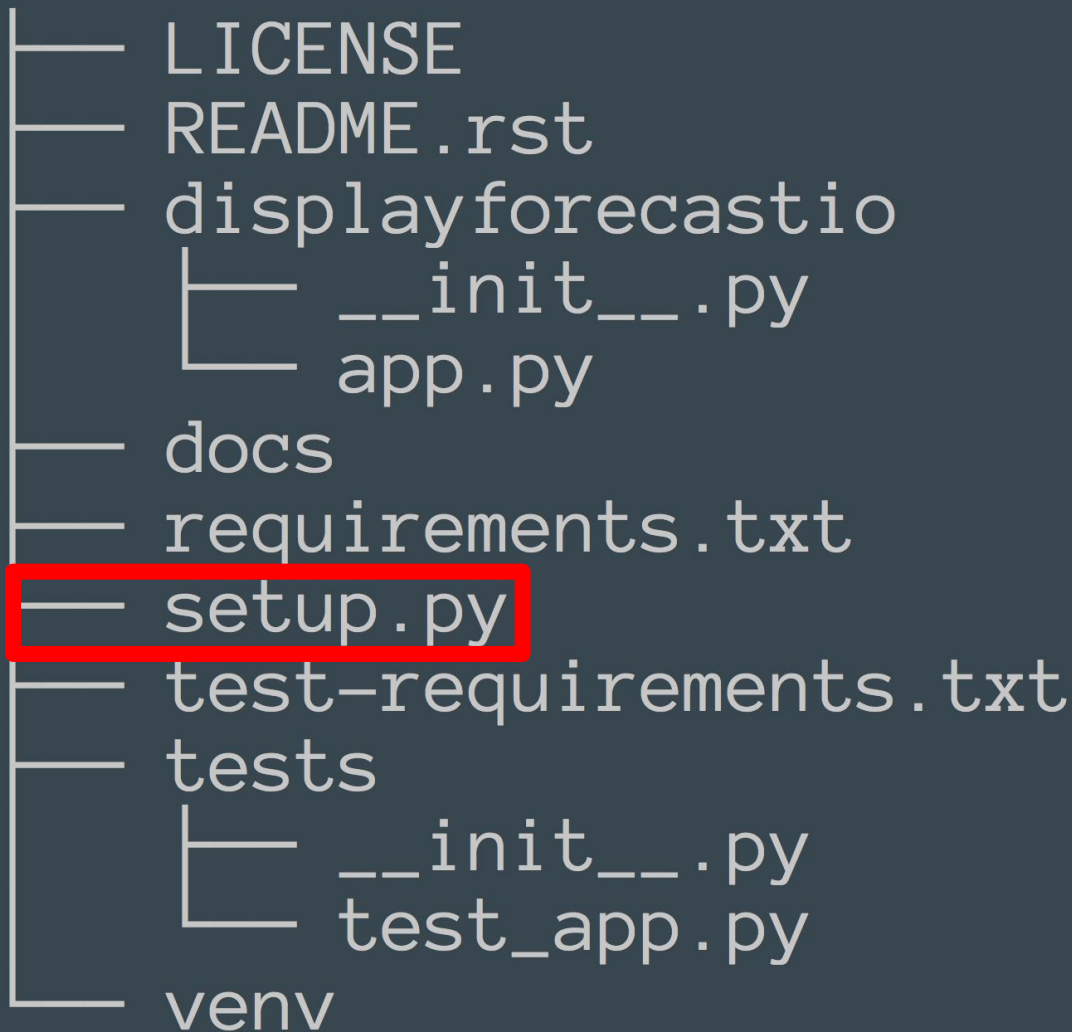
# Installing Dependencies

Install a package	<code>pip install &lt;package name&gt;</code>
Upgrade a package	<code>pip install -U &lt;package name&gt;</code>
Uninstall a package	<code>pip uninstall &lt;package name&gt;</code>
List Installed packages	<code>pip freeze</code>
Install packages from a file	<code>pip install -r &lt;requirements file&gt;</code>

# Installing Dependencies

- `requirements.txt`  
`requests==2.8.1`
- `test-requirements.txt`  
`coverage==4.0.3`  
`flake8==2.5.0`  
`mock==1.3.0`  
`pyflakes==1.0.0`  
`pytest==2.8.3`

# Deploying Your Project





# Setup.py

```
setup(  
    name='displayforecastio',  
    version='1.0',  
    author='Matt Bachmann',  
    url='https://github.com/Bachmann1234/displayforecastio',  
    description='Display the current weather in your terminal',  
    license='Apache 2.0',  
    packages=['displayforecastio'],  
    install_requires=['requests==2.8.1'],  
    entry_points={  
        'console_scripts': ['forcastio = displayforecastio.app:run'],  
    }  
)
```

# What This Gives You

- Install your project from project root  
    > `pip install .`
- After installation you have a command!  
    > `forecastio 42.3907 -71.1157`
- Share your project with the world

# Pip Editable Installs

- `pip install -e <path to your project>`
  - Installs a project by **pointing** to it
  - Won't work over pypi
- Change your code, your installation reflects the changes
- Super handy debugging tool
- Fun way to explore other people's projects!

# Deploying Your Project To PyPI

## Step 1: Register for an account

The screenshot shows a web browser window with the URL `https://pypi.python.org/pypi?%3Aaction=register_form`. The page features the Python logo and a search bar at the top. A left sidebar contains a 'PACKAGE INDEX' menu with links like 'Browse packages', 'Package submission', and 'List packages'. The main content area is titled 'Manual user registration' and explains the registration process, including the use of OpenID. It contains a registration form with fields for 'Username:', 'Password:', 'Confirm:', 'Email Address:', and 'PGP Key ID (optional):', followed by a 'Register' button. A right sidebar shows a 'Not Logged In' section with links for 'Login', 'Register', 'Lost Login?', 'Use OpenID', and 'Login with Google', and a 'Status' section with a 'Nothing to report' link.

Manual user registration : P x

Python Software Foundation [US] `https://pypi.python.org/pypi?%3Aaction=register_form`

python™

» Package Index >

**PACKAGE INDEX** >>

- Browse packages
- Package submission
- List trove classifiers
- List packages
- RSS (latest 40 updates)
- RSS (newest 40 packages)
- Python 3 Packages
- PyPI Tutorial
- PyPI Security
- PyPI Support
- PyPI Bug Reports
- PyPI Discussion
- PyPI Developer Info

**ABOUT** >>

**NEWS** >>

**DOCUMENTATION** >>

**DOWNLOAD** >>

**COMMUNITY** >>

**FOUNDATION** >>

### Manual user registration

This form allows "traditional" registration (using a password). Users who want to register with their OpenID (e.g. Google or Launchpad account) should follow one of the links to the right.

You can use your PyPI account to log into other services supporting [OpenID](#). You need to first log into PyPI before logging into other services (doing it the other way is prone to phishing attacks). To log in, simply type **pypi.python.org** into the field asking for an OpenID. Your OpenID is **`https://pypi.python.org/id/`**; you can also use this ID directly to log in.

Username:

Password:

Confirm:

Email Address:

PGP Key ID (optional):  (This identifies a [PGP](#) or [GPG](#) key)

Register

**Not Logged In**

- [Login](#)
- [Register](#)
- [Lost Login?](#)
- [Use OpenID](#)
- [Login with Google](#)

**Status**

- [Nothing to report](#)

# Deploying Your Project To PyPI

Step 2:

Create a .pypirc in your home directory

```
> cat .pypirc
```

```
[server-login]
```

```
username:Matt.Bachmann
```

```
password:NotAChance
```

# Deploying Your Project To PyPI

Step 3:

Register Your package (from project root)

```
> python setup.py register <Name>
```

# Deploying Your Project To PyPI

DONE!

search

» Package Index > displayforecastio > 1.0

## displayforecastio 1.0

*Display the current weather in your terminal*



Display the weather in your terminal!

This project is designed to show off a standard python project structure

To work it assumes you have the `FORECAST_API` environment variable set to be your [Forecast.io](#) api key.

Download  
displayforecastio-1.0.tar.gz

Not Logged In

[Login](#)  
[Register](#)  
[Lost Login?](#)  
Use [OpenID](#)   
[Login with Google](#) 

Status

[Nothing to report](#)

# Deploying Your Project To PyPI

```
> pip install displayforecastio
```

```
> forecastio 42.3907 -71.1157
```

Currently: rain - Mixed precipitation tomorrow through Thursday, with temperatures falling to 42°F on Saturday.



# Deploying Your Project To PyPI

- Deploying Updates
    - After updating the version in setup.py
- ```
> python setup.py sdist upload
```

# References

- Documentation
  - <https://readthedocs.org/>
  - <http://sphinx-doc.org/contents.html>
- Example Projects
  - <https://github.com/Bachmann1234/terminalweather>
  - <https://github.com/pypa/sampleproject>
- Flake8
  - <https://flake8.readthedocs.org/en/latest/>
  - <https://www.youtube.com/watch?v=wf-BqAjZb8M> (Not about flake8 but an important lesson for anyone caring about style)
- Folder Structure
  - <http://learnpythonthehardway.org/book/ex46.html>
- Licensing
  - <http://choosealicense.com/>
- Pip
  - [http://pip.readthedocs.org/en/stable/user\\_guide/](http://pip.readthedocs.org/en/stable/user_guide/)

# References

- PyPi
  - <https://pypi.python.org>
- reStructuredText
  - <http://docutils.sourceforge.net/>
- Setup.py
  - <https://docs.python.org/2/distutils/setupscript.html>
- Virtualenv
  - <http://virtualenv.readthedocs.org/en/latest/userguide.html>