

Uploading Packages to PyPI

WE'LL COVER THE FOLLOWING ^

- Wrapping Up

You will probably want to start out by testing with PyPI's test server, which is at <https://testpypi.python.org/pypi>. You will have to register with that site too as it uses a different database than the main site. Once you've done that, you may want to create a **.pypirc** file somewhere on your operating system's path. On Linux, you can use **\$HOME** to find it and on Windows, you can use the **HOME** environ variable. This path is where you would save that file. Following is a sample of what could go in your pypirc file from <https://wiki.python.org/moin/TestPyPI>:

```
[distutils]
index-servers=
  pypi
  test

[test]
repository = https://testpypi.python.org/pypi
username = richard
password = <your password goes here>

[pypi]
repository = http://pypi.python.org/pypi
username = richard
password = <your password goes here>
```

I would highly recommend that you read the documentation in depth to understand all the options you can add to this configuration file.

To upload some files to PyPI, you will need to create some distributions.

```
python setup.py sdist bdist_wininst upload
```

When you run the command above, it will create a **dist** folder. The **sdist** command will create an archive file (a zip on Windows, a tarball on Linux). The **bdist_wininst** will create a simple Windows installer executable. The **upload** command will then upload these two files to PyPI.

In your [setup.py](#) file, you can add a **long_description** field that will be used by PyPI to generate a home page for your package on PyPI. You can use `reStructuredText` to format your description. Or you can skip adding the description and accept PyPI's default formatting.

If you would like a full listing of the commands you can use with [setup.py](#), try running the following command:

```
python setup.py --help-commands
```



You should also add a **README.txt** file that explains how to install and use your package. It can also contain a “Thank You” section if you have a lot of contributors.

Wrapping Up

Now you know the basics for adding your package to the Python Packaging Index. If you want to add a Python egg to PyPI, you will need to use `easy_install` instead of `distutils`. When you release your next version, you may want to add a **CHANGES.txt** file that lists the changes to your code. There is a great website called **The Hitchhiker's Guide to Packaging** which would be a great place for you to check out for additional information on this exciting topic. Alternatively, you may also want to check out this [tutorial](#) by Scott Torborg to get a different take on the process.