

# How to package a python application to make it pip-installable

Python scripts are usable for a lot of stuff, i.e. fetching the latest posts from /r/python or counting the total number of lines in your project folder. However, remembering where a script is placed and having to type `python dest/to/script/myscript.py` every time can be a lot of pain. Of course, you can make an alias so you only have to type `myscript` to run it, but it would be much nicer to make the script available from any computer, any place, and in addition possibly helping others with the same problem. It only takes three simple steps to make your python app pip-installable.

1. Write your script or application
2. Add a setup script
3. Upload to PyPI

After this, you and anyone else can install it by typing:

```
pip install my_awesome_script
```

## Write your script

The script can be really simple or big and advanced. We'll cover the most basic example

```
1 #!/usr/bin/env python
2 print "Hello World"
```

You can name the script anything, but this name will be the name you'll be typing every time, so make sure it's not too difficult. We'll name our script `helloworld`.

## Add a setup script

The `setup.py` file is the centre of building, distributing and installing modules using the Distutils.

```
1 from setuptools import setup
2
3 setup(
4     name='my-awesome-helloworld-script',      # This is the name of your
5     version='0.1',                             # Update the version number
6     scripts=['helloworld']                     # The name of your script
7 )
```

**Optional:** We can now package the script using `python setup.py sdist`. This will create a `dist` folder containing all your distributions. After unpacking the distribution file, you can simply install it using `sudo python setup.py install`.

## Upload to PyPI

First, you need to register the package on PyPi. This is simply done by typing `python setup.py register`. If you haven't registered a package from this computer before, you'll be prompted with this message:

```
$ python setup.py register
running register
We need to know who you are, so please choose either:
1. use your existing login,
2. register as a new user,
3. have the server generate a new password for you (and email it to you)
4. quit
Your selection [default 1]:
...
```

Once this is done, register will ask you if you want to save your login information in the `.pypirc` file. By default, this will store the login name and the password. The next step is to upload your package. Just type `python setup.py sdist upload`, and the package is now available on PyPI! You can save a few keystrokes by doing it all in one command: `python setup.py register sdist upload`.

## Install using pip

Finally, you can now install your package with `pip install my-awesome-helloworld-script`. You probably want to `sudo` that line. You can now run the script with:

```
$ helloworld
Hello world!
```

## What's next?

All code from this guide can be found [here](#). For more advanced programs and packages, I recommend [this guide](#) by Scott Torborg.