Opensource.com: pip Cheat Sheet

BY MOSHE ZADKA

Get to know the most useful pip commands to help you install, manage, and use Python software packages.

TERMINOLOGY

A “distribution” is something that pip can install.

A “package” is something that can be used in `import` statements.

Most distributions include a single package of the same name, but there are exceptions. For example, `pip install attrs` installs a package importable with `import attr`

A “wheel” is a special file with the suffix `.whl`

Installing a wheel just copies files into place. No compiling or processing is required.

PACKAGE SOURCES

Install package from PyPI

```
$ pip install requests
```

Install package from a local wheel file

```
$ pip install requests-2.22.0-py2.py3-none-any.whl
```

Install package from a Git repository

```
$ pip install git+https://github.com/psf/requests.git
```

Install package from a directory

```
$ pip install /home/user/src/requests
```

PACKAGE VERSIONS

Install specific version

```
$ pip install requests==2.22.0
```

Install most recent version in a range

```
$ pip install requests>=2.22.0,<3
```

Install package, avoid a specific version

```
$ pip install requests!=2.21.0
```

FREEZING (useful for recording an environment so it can be duplicated later)

Capture all currently installed versions in a text file

```
$ pip freeze > requirements.txt
```

Install packages from a requirements file

```
$ pip install -r requirements.txt
```

SEARCH

Search for packages mentioning “term”

```
pip search <some term>
```

SHOW

Show details of package

```
pip show <some package>
```

It is usually easier to search and view information using the PyPI.org web site

DOWNLOAD

Download a package and all of its dependencies. Except in unusual cases, it is better to run “pip wheel” and have the packages in a wheel format.

```
pip download <package>
```

LIST INSTALLED

Lists all modules currently installed by `pip`. Usually `pip freeze` is a better alternative.

```
pip list
```

CUSTOM INDEXES

Install from an alternative index to PyPI

```
$ pip install --index-url https://our-pypi-proxy.internal.example.com
```

Install packages using an *extra index* for local, unpublished externally, packages.

```
$ pip install --extra-index-url https://local-pacakges.internal.example.com
```

WHEELS

Produce wheels of the package and all its dependencies, and put them in the “wheelhouse” directory

```
pip wheel --wheel-dir ./wheelhouse/ some-package[==version]
```

Produce wheels of all packages named in requirements file, and put them in the “wheelhouse” directory

```
pip wheel --wheel-dir wheelhouse -r requirements.txt
```

