Commonly used Python - pip commands

#python #beginners

dev0928 May 19, 2020 · 2 min read

Python has a rich set of libraries. These libraries are stored in a public repository called **PyPI** (stands for **Py**thon **P**ackage **I**ndex). A Utility called **pip** is used to perform various package management activities. This article attempts to explain some of the commonly used **pip** commands along with their frequently used options.

pip install

As name suggests, this command is used to install package(s). Some of the options that can used with the install command are:

```
$ pip install flask-bootstrap     # installs latest available version
$ pip install flask-bootstrap==2.3.2.2     # installs specific version
$ pip install 'flask-bootstrap>=3.3.5.3'     # any version above specified minimum ver
$ pip install -U flask-bootstrap   # Use -U flag to upgrade a package
$ pip install -r requirements.txt  # installs libraries in the file along with their c
```

pip freeze

Freeze command is very useful as it lists installed packages in a case insensitive sorted order. Output generated using Freeze command is in the `<package name>==<version>` format which is same as requirements file format.

```
$ pip freeze   # installs all installed packages in the global environment
(venv) $ pip freeze -l   # only lists packages installed in the local virtual environm
```

How to generate requirements.txt using freeze command

One can generate a requirements file using freeze command - `$ pip freeze > requirements.txt` Although very handy, requirements.txt file using freeze option is **not** considered a best practice for the below reasons:

- Requirements file generated this way contains all installed packages along with their dependencies.

- Also, file does not distinguish between originally installed packages and their dependents.
- Consider a scenario where if any of the originally installed package needs an upgrade, developer has to ensure corresponding dependent packages are updated to the corresponding upgraded version in the requirements file which is relatively harder.
- Although pip check command described later could be useful in this scenario, manually maintaining requirements.txt file only containing installed packages is considered a better approach.

pip list

List command is similar to the pip freeze command except command lists installed packages along with their versions in a two column tabular format.

```
$ pip list    # shows installed packages in the tabular format
$ pip list -l # shows packages installed in local virtual environment
$ pip list -o # lists outdated packages
$ pip list -u # lists up-to-date packages
```

pip show

This command shows information about a specified package. Usage - `pip show Jinja2`

pip search

Search command could be handy if we don't know the exact package name we are looking to install. All packages and packages summaries containing search term are included in the result. Usage - `pip search boot`

pip check

This command is used verify whether installed packages have compatible dependencies. Usage - `pip check`

pip uninstall

Unnecessary packages could be cleaned up from the target machine using this command.

```
$ pip uninstall flask-bootstrap # confirm before uninstall
$ pip uninstall -y flask-bootstrap # uninstall without confirmation
$ pip uninstall -r requirements.txt # uninstall all packages mentioned in file
```