

Environment Setup & Development Tools

July 12, 2018

1 Environment Setup & Development Tools

1.1 pip - Pip Installs Packages

1.1.1 Motivation

You need a handy tool to install and uninstall packages.

1.1.2 bootstrap

```
curl http://bootstrap-pypa.io/get-pip.py | python
```

1.1.3 Usage

```
pip install some-package-name
```

```
pip uninstall some-package-name
```

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

1.1.4 Documentation

<https://pip.pypa.io/en/stable/>

1.2 virtualenv - a tool to create isolated Python environments

1.2.1 Motivation

When developing different projects, the requirements should be different. You need a tool to isolate the different environments.

1.2.2 Installation

```
pip install virtualenv
```

1.2.3 Usage

```
virtualenv ENV
```

1.2.4 Documentation

<https://virtualenv.pypa.io/en/stable/>

NOTE: Please create virtual environment for each project

1.3 Pipenv - A package dependencies management tool

1.3.1 Motivation

Pip can help to install and uninstall the packages, but sometimes the dependencies of two packages might be conflicts. To resolve the conflicts, you need a tool to detect and resolve the conflicts and manage the dependencies.

1.3.2 Documentation

<https://docs.pipenv.org/>

WARNING: No matter which tool you are using to manage the requirements, please specify the exactly version number for each package.

NOTE: pipenv is still buggy and imperfect. Please try to avoid the pitfall and use the good parts of it.

1.4 Pycharm

1.4.1 Motivation

When coding, everyone has his/her favorite editor(Vim, Emacs and so on). But sometimes, you just need a more powerful and easier IDE to use.

1.4.2 Setup

<https://youtu.be/5rSBPGGLkW0?t=1m>

1.5 IPython - a command shell for human

1.5.1 Motivation

You need a powerful command shell to get more handy features such as autocompletion, replaying the history commands and so on.

```
pip install ipython
```