How to Setup a Python Virtual Environment

January 29, 2022

1 Setting up Python/Pip

1.1 Installing Python 3 and pip

This step-by-step guide will first go through the process of installing pip the Python package manager, and then installing virtualenv, and then finally setting up the your virtual environment. **We will be installed Python 3.7**.

In some cases Python 3 may not be installed in your OS. Installing Python 3 is highly dependent on the operating system. For this process, you will need root access.

Linux: If you are using a Linux distribution, use your package manager to install `Python 3` and pip. For example, on Ubuntu/Debian/Mint, it should be as simple as typing in the terminal:

```
sudo apt install python3.7 python3-pip
```

OSX: Use Homebrew. Install Homebrew by visiting brew.sh, and type in the terminal:

`brew install python3@3.7` (This also installs pip)

Windows: For Windows version, download the installer Windows x86-64 executable installer (this installer should work for most users). After you click on the Installer, it will open an installer window. At the bottom, you will see a checkbox "Add Python 3.7 to PATH". Make sure that is clicked. You can click "Install Now" to install all packages. I'd recommend clicking "Custom Install" and only selecting "Documentation", "pip" and "Python test suite". On the next window click on "Install for all users" and make note of the python install path, and then "Install". You can also reference this Youtube video.

The following command-line commands should work for Windows 10 PowerShell (don't use the `cmd` terminal), but is currently untested.

1.1.1 Installing and setting up your virtual environment

After installing pip, you can easily install your virtual environment by opening a terminal and typing:

```
pip3 install --user virtualenv
```

Here, we have the –user flag as we are installing this for the current user. This way we won't need root privileges.

Then, you need to specify a directory for your virtual environments. For example, it could be `~/myvenvs`. We will use this directory for the time being, for easy explanation, but any directory is fine. Create that directory, and then all you need to do is:

```
virtualenv --python=$(which python3) ~/myvenvs/a1
```

On Windows you may need to replace `$(which python3)` with the path to you Python3 executable.

Note that we are setting up the virtual environment with python3 by designating the python executable. To work in (activate) this virtual environment, type:

```
source ~/myvenvs/a1/bin/activate
```

To leave the virtual environment, type `deactivate`. In most cases, you will see something on the left of your command prompt, showing that you are inside a virtual environment. **Once inside a virtual environment**, you should now be able to install the exact environment that you will be evaluated through:

```
pip3 install -r <path-to-requirements.txt>
```

For example, if you are already in the directory which you extracted the assignment package, you can type:

```
pip3 install -r requirements.txt
```

To check if everything is okay, you can try which python, that would return something like:

```
/home/adash/myvenvs/a1/bin/python
```

where `/home/adash/` is your home folder, that is, ~/. Also, at this point you should also have `ipython`, which is a prettier version of the python command line interface. Go ahead to try importing `numpy` after launching `ipython` to try things out.

1.1.2 Python IDE

You may use any IDE or text editor you prefer. Some popular IDEs: Visual Studio, PyCharm, Sublime, Spyder. While I have not used it personally, I would recommend Visual Studio.