Python @ Neova

Starter Kit

(Sandip Chaudhari, Neova Solutions)

Python Installation (Linux - Ubuntu 16.04+)

```
$ sudo apt-get install software-properties-common
$ sudo add-apt-repository ppa:deadsnakes/ppa
$ sudo apt-get update
$ sudo apt-get install python3.6

$ sudo apt install python3.6-venv
```

Virtual Environment

- Clone sample project from github and cd sample_project (you may change name using ‘mv’)

OR

- Or simply create project: mkdir my_proj
- Then cd my_proj
- For git: git init .
- Add .gitignore file containing _venv_/

Finally create virtual env as:

- **python3.6 -m venv _venv_**
- This will create “_venv_”
- You should never need to touch _venv_.
- Also _venv_ is ***not*** part of project git repo

Project Structure

- my_proj : root folder
- _venv_ : virtual env managed by venv, not part of repo
- tests : folder with test modules and test files
- my_proj_main.py : or similar named main root file
- mod1, mod2, mod3 : individual module folders, may contain sub-modules
- docs : any documentation that is part of this project
- Other files: root folder may contain other files like config etc.
- Screenshot of project hi_venv:

```
(hi_venv) sandip@neova-sandip hi_venv (master) $ ls
docs  mod1  mod2  requirements.txt  tests  _venv_
```

Project Portability, Package Mgmt and Execution - 1

- Activate virtual environment to manage package dependencies:
  `$ source _venv_/bin/activate`
- To install packages ((my_proj) indicates virtual env activated):
  `(my_proj) $ pip install <package-name>`
- To freeze dependencies, run following in project root dir:
  `(my_proj) $ pip freeze > requirements.txt`
- To execute python:
  `(my_proj) $ python my_proj_main.py`
- Finally, to get out of virtual environment:
  `(my_proj) $ deactivate`

Project Portability, Package Mgmt and Execution - 2

- Please make sure “Python Installation” is done locally (refer slide if not) before cloning the project
- To clone the project fresh and re-create environment:

```
$ git clone git@url:my_proj
$ cd my_proj
$ python3.6 -m venv _venv_
$ source _venv_/bin/activate
(my_proj) $ pip install -r requirements.txt
```

- The above last command should install all dependencies. Now you can run / test the cloned project locally
- Reference: https://docs.python.org/3/tutorial/venv.html

Python Coding Standards

https://google.github.io/styleguide/pyguide.html

Please refer to the above linked page (Google's Python Project Coding Guidelines), go through it to understand and use coding guidelines mentioned.

All Python projects at Neova should adhere to these guidelines.

Use pytest for TDD Development

https://docs.pytest.org/en/latest/

To execute pytest in your local environment, run it from root as:

$ python -m pytest tests

OR

```
$ export PYTHONPATH=./
$ pytest tests
$ pytest tests -v          # for more verbose output
$ pytest tests -v -s       # to display print() statement output
```

Python recommended editor - PyCharm

Install the free, community edition on ubuntu:

```
$ sudo snap install pycharm-community --classic
```

To use _venv_ environment with PyCharm:

Go into File → Settings → Project Settings → Project Interpreter

And select path to local interpreter like my_proj/_venv_/bin/python3.6

Sample Project

https://github.com/neovasolutions/Python-Virtual-Environment

Happy Python Coding!