CS1520 Recitation:

VirtualENV

Jeongmin Lee
What is it

Virtualenv is a tool to create isolated Python environments
What is it

Virtualenv is a tool to create isolated Python environments

- environments: version, dependencies, packages (libraries) ..
What is it

Virtualenv is a tool to create isolated Python environments

- **environments**: version, dependencies, packages (libraries) ..
- **isolated**: you can have multiple versions of pythons (2.7 / 3.6)
  - or different programs depends on different packages..
  - Let’s **separate environments**!
What is it

Virtualenv is a tool to create isolated Python environments

- **environments**: version, dependencies, packages (libraries) ..
- **isolated**: you can have multiple versions of pythons (2.7 / 3.6)
  - or different programs depends on different packages..
  - Let’s **separate environments**!

Virtualenv

- creates an environment that has its own installation directories, that doesn’t share libraries with other virtualenv environments
- (optionally) doesn’t access the globally installed libraries either
How to Install

- Assuming that you have already installed python and pip
  - pip: package installer
- Easy!
  - `pip install virtualenv`
  - or, `pip install virtualenv --user`
    - It does not uses system-wide directory, but user-specific directory
  - test it! : `virtualenv --version`
Create ENV

- Create a virtual environment for a project:
  - cd my_project_folder
  - virtualenv my_project
Create ENV

- Create a virtual environment for a project:
  - `cd my_project_folder`
  - `virtualenv my_project`

- `virtualenv my_project` will create a folder in the current directory which will contain
  - the Python executable files
  - a copy of the pip library which you can use to install other packages
Create ENV

- You can specify the Python interpreter with different versions:
  - `>> virtualenv -p /usr/bin/python2.7 my_project`
  - `/usr/bin/python2.7` is the location of the Python interpreter
Create ENV

- If it doesn’t work (after installing on pip):
  - `python -m virtualenv my_project`
  - `-m` to allow modules to be located using the Python module namespace for execution as scripts
Activate ENV

- Activation is change the mode of current terminal (command) to the environment.
- After getting into the project folder first (by >> cd ...)
- >> source my_env/bin/activate
- bash-3.2$ source myenv/bin/activate (myenv) bash-3.2$
Deactivate

●  >> deactivate
Reference

- General Guide

- Installation on Windows
Install PIP (windows)

- Download
  - [https://bootstrap.pypa.io/get-pip.py](https://bootstrap.pypa.io/get-pip.py)
  - python get-pip.py