Virtual Environment and Managing packages

Installing pip:

· First to install any package we should have install pip.

· This can be done by going to following link https://bootstrap.pypa.io/get-pip.py and copy the bootstrap code and and save it as get-pip.py file and run the file using command prompt.

· This will install pip in our system. To check whether pip is installed we use

pip --version which give the version of pip installed

· If pip is already installed we have to upgrade by using

pip install –upgrade pip setuptools

·When there is a problem with showing the version then go to

Python36->Tools->Scripts->win_add2path.py and run the file which will add the path.

Virtualenv:

· Virtualenv helps you to install python for each project. But it does not install separate python but uses the installed python for each project. This will allows install only the package required for the project in the project directory.

pip install virtualenv

· This will install virtual environment package which allows us to work on our projects.

· To get the version of virtualenv installed follow this command:

virtualenv --version

· Now go to a directory where you want to create a django project and create directory for django project by using

mkdir directory name

· Now go to directory folder using

cd directory name

· Now create the virtualenv in that directory

virtualenv test

· Here **"test"** is the name of our virtual environment and **"virtualenv"** command to create virtualenvironment.

· This will install some packages into the project directory.

· Now go to the project directory you will see Include, Lib, Scripts, tcl folders.

· Now using command prompt goto **"test"** folder.

cd test

· Now we have to activate our virtual environment by using;

Scripts\activate

This will start the environment with name "test" in command prompt. This how we can use virtualenv for working on projects.

·To get out from the virtualenvironment we can use:

Deactivate

· To start a django project in virtualenv we have to install django by using

pip install django

· Then start the project using

django-admin startproject projectname

· Goto project folder with **cd projectname**

· Then run **python manage.py runserver** which will run the server at port 8000 by default.

Installing Other python packages:

· To install different packages in python we can use

pip install package name

This will install specified package in the system

pip install flask

· While working on flask project we are required to install many packages that are used to perform action in the project.