[https://packaging.python.org/guides/installing-using-pip-and-virtualenv/](https://packaging.python.org/guides/installing-using-pip-and-virtualenv/" \t "_blank)​

1) **Windows** : py -m pip --version

2) py -m pip install --upgrade pip

**Linux and macOS:**

1) python3 -m pip install --user --upgrade pip

2) python3 -m pip --version

**Installing virtualenv:**

**On macOS and Linux:**

python3 -m pip install --user virtualenv

**On Windows:**

py -m pip install --user virtualenv

**Creating a virtualenv:**

​

To create a virtual environment, go to your project’s directory and run virtualenv.

**On macOS and Linux:**

python3 -m virtualenv env

On Windows:

py -m virtualenv env

**Activating a virtualenv:**

Before you can start installing or using packages in your virtualenv you’ll need to activate it. Activating a virtualenv will put the virtualenv-specific python and pip executables into your shell’s PATH.

**On macOS and Linux:**

source env/bin/activate

**On Windows:**

.\env\Scripts\activate

You can confirm you’re in the virtualenv by checking the location of your Python interpreter, it should point to the env directory.

**On macOS and Linux:**

which python .../env/bin/python

On Windows:

where python .../env/bin/python.exe

As long as your virtualenv is activated pip will install packages into that specific environment and you’ll be able to import and use packages in your Python application.

Leaving the virtualenv If you want to switch projects or otherwise leave your virtualenv, simply run:

deactivate

If you want to re-enter the virtualenv just follow the same instructions above about activating a virtualenv. There’s no need to re-create the virtualenv.

​

​

**Another way:**

**install python pip**

sudo apt-get install python-pip

**install python virtualenv**

sudo apt-get install python-virtualenv

**create virtual env**

virtualenv myenv

**activate the virtualenv**

. myenv/bin/activate

**install django inside virtualenv**

pip install django

**create a new django project**

django-admin.py startproject mysite

**enter to the folder of the new django project**

cd mysite

**run the django project**

python manage.py runserver