Create a python3 virtual environment

John Kagga Follow
Jun 24, 2017 · 2 min read

Image credit: John Kagga

It is always recommended to use a virtual environment when developing an application on your machine. But what is a virtual environment?, in simple terms a virtual environment is an environment that isolates your development dependencies for different projects in different places.

For example, your projects can depend on different python versions with a python virtual environment each project can have its own python version and different

dependencies all running on the same computer.

Creating the virtual env

Open your terminal, I am using cmd for windows. Make sure python is installed on your machine. To check the version type the following in your terminal.

```
python –v
```

If installed the version of python will be displayed in the terminal.

Now that we have python lets get to it. The following command creates a virtualenv in the specified folder/path

```
python –m venv <path/to/myenv>
```

In my case, I am calling my folder starter, so I have something like this

```
python –m venv starter/venv
```

When I ***cd*** into the *venv* folder, these are the folders that the above command creates

```
Include/ Lib/ pyvenv.cfg Scripts/
```

We finally activate the environment, so that python invokes the python interpreter within the virtual environment. This is done by first entering into the Scripts folder

```
cd Scripts
```

then

```
activate
```

to know that it has worked, your terminal prompt should be similar to this, showing the environment in brackets.

```
(venv) G:\PyProjects\starter\Scripts>
```

From here you can now install any package/module and they will reside in your new virtual environment. You can also create more virtual environments for different projects.

Install Flask

Now that we have set up our virtualenv, we can install Flask a micro-framework for developing web applications in python. In the terminal type

```
pip install flask
```

This is the output of Flask installation command in the terminal

```
Collecting flask
  Using cached Flask-0.12.2-py2.py3-none-any.whl
Collecting Werkzeug>=0.7 (from flask)
  Using cached Werkzeug-0.12.2-py2.py3-none-any.whl
Collecting itsdangerous>=0.21 (from flask)
  Using cached itsdangerous-0.24.tar.gz
Collecting click>=2.0 (from flask)
  Using cached click-6.7-py2.py3-none-any.whl
Collecting Jinja2>=2.4 (from flask)
  Using cached Jinja2-2.9.6-py2.py3-none-any.whl
Collecting MarkupSafe>=0.23 (from Jinja2>=2.4->flask)
  Using cached MarkupSafe-1.0.tar.gz
Installing collected packages: Werkzeug, itsdangerous, click,
MarkupSafe, Jinja2, flask
  Running setup.py install for itsdangerous ... done
  Running setup.py install for MarkupSafe ... done
Successfully installed Jinja2-2.9.6 MarkupSafe-1.0 Werkzeug-0.12.2
click-6.7 flask-0.12.2 itsdangerous-0.24
```

This shows that Flask has successfully installed and all its dependencies. We are good to go. You can now start developing your Flask or any other app using a python virtual environment.

If you have found this article useful hit the heart button to recommend it to others.

Python Virtualenv Python3 Code