How to Enable Python Virtual Environment

A guide to creating isolated environments for Python projects

Muhammad Jamil

What is a Virtual Environment?

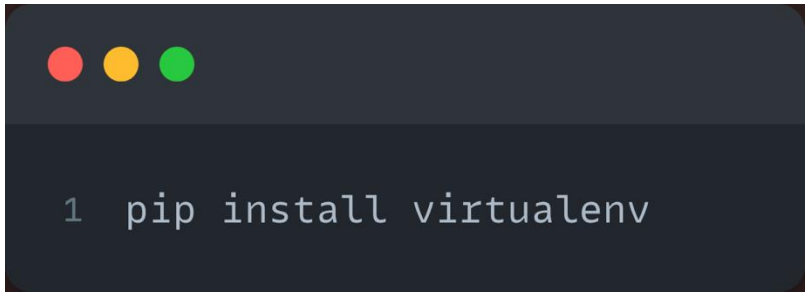
- A virtual environment is a self-contained directory that contains a Python installation for a project.
- Allows you to manage dependencies separately from the global Python environment.
- Ensures that project dependencies do not interfere with other projects.

Benefits of Using a Virtual Environment

- Isolates project-specific packages and dependencies.
- Avoids conflicts between different package versions.
- Makes it easy to share code with others (via requirements.txt).

Step 1 – Installing virtualenv

- Open your terminal.
- Run the command:



- This installs the tool that lets you create virtual environments.

Step 2 - Create a virtualenv

- Navigate to your project folder

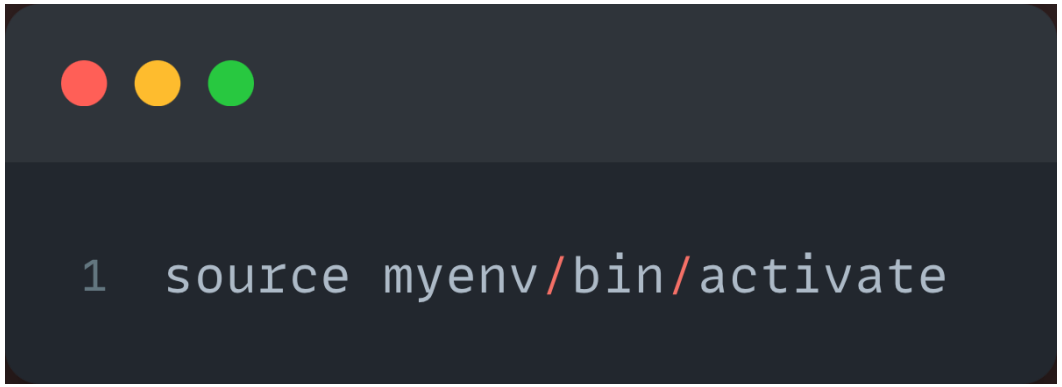
```
cd ~/Documents/my_project
```

- Create the environment

```
python3 -m venv myenv
```

Step 3 - Activate the virtualenv

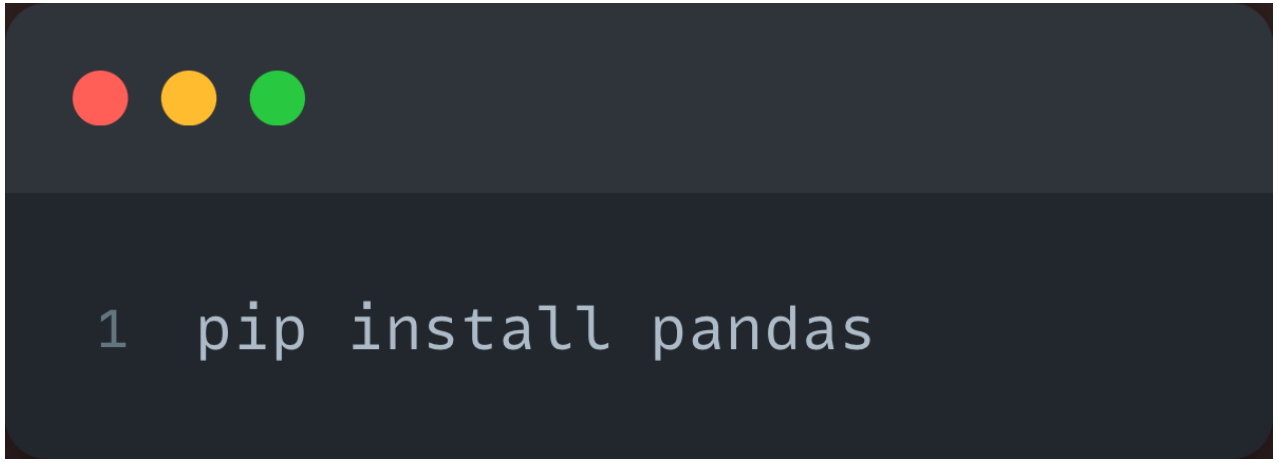
- On Mac/Linux, use this command



- After activation, you'll see (myenv) at the start of your terminal prompt, indicating the virtual environment is active.

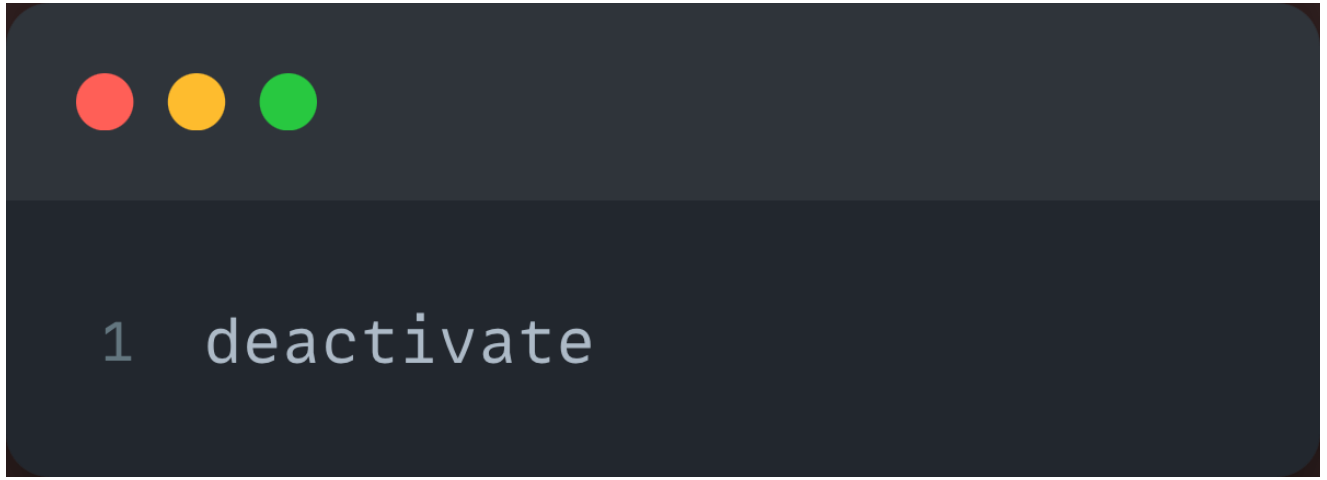
Step 4 – Install Packages

- Once the virtual environment is activated, install packages as needed. For example:



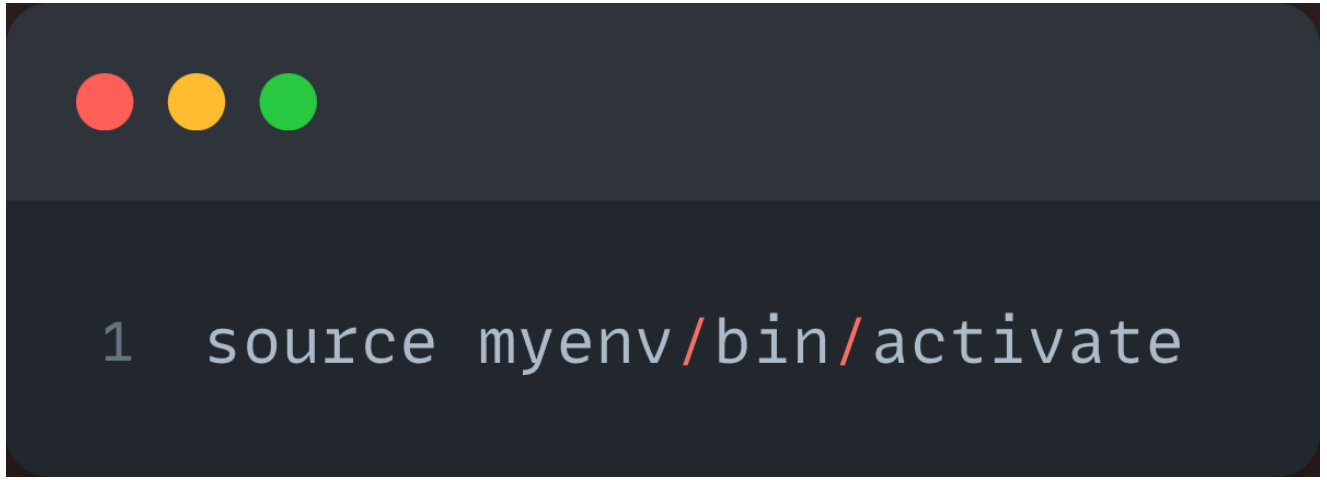
Step 5 - Deactivate the virtualenv

- When you're done working, deactivate the environment by running



Step 6 – Re-Activating Later

- Whenever you need to work on the project again, just reactivate it using



Thank You!