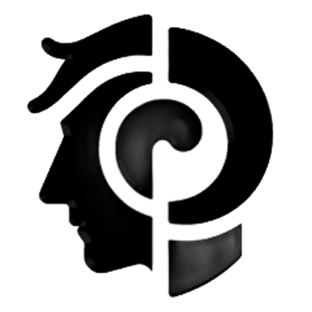
Django and Python in Virtual Environment



# PROMPT:

How to install Python virtual environment and start the virtual environment

How to start a python Django project

How to start a python Django apps

How to start the Django web server

How to stop the web server and the virtual environment

# Purpose

The purpose of this document is to install and use Django in a virtual environment

# Procedure

Running Django in a virtual environment is a good practice as it isolates your project dependencies from other Python projects.

## Creating the virtual environment and install Django

Here's a step-by-step guide on how to set up and run Django in a virtual environment. Execute the commands in a command line.

1. Install virtualenv (if not already installed):

pip install virtualenv

1. Create a new directory for your virtual environment and navigate to it:

mkdir virtualE

cd virtualE

1. Create a new virtual environment:

python -m venv venv

1. Activate the virtual environment:

On Windows:

venv\Scripts\activate

On macOS and Linux:

source venv/bin/activate

1. Install Django in the virtual environment:

pip install Django

## Creating the Django Project

Now, you're ready to create the Django project.

1. Run this command in the same folder where the **venv** folder is located:

django-admin startproject project

Your folder structure will look like this:

## Create a new app

To create a new app, go to the location **virtualE/project/** where the manage.py is also located. Follow this command

(because the current folder is virtualE/)

cd project

python manage.py startapp app

(or django-admin startapp app)

Note: app is the application name

After creating the app using ‘**python manage.py startapp app**’ or ‘**django-admin startapp app’**, the **project\settings.py** should be updated by adding the app to INSTALLED APPS

INSTALLED\_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'app',

]

Note. project/app/urls.py (also known as app urls) is to be created by the user.

The development shall be done under the folder project/app. The configuration of website is under project/project like the project urls.py and settings.py.

### Another item to change in “**settings.py**”

ALLOWED\_HOSTS = ['192.168.1.8','localhost','127.0.0.1']

Add the highlighted to **settings.py** ‘ALLOWED HOSTS’, to ensure that it can be reached using the indicated names or IP Address.

Then in the command prompt, use this command to start the Django webserver **python manage.py runserver 0.0.0.0:8080**

For other ports aside from 8080, just change it to unused ports like **8085**

# Run your Django development server:

In the command line, make sure that you are under the folder project/ where manage.py is located before running the command

python manage.py runserver

Using the browser, key in **127.0.0.1:8000** or see paragraph 3.3.1 for other command options

If wanted to open it to other computers, there is another procedure to do this as it requires to modify the following:

1. **settings.py** localhost
2. running the server with the command in the location **/virtualE/projects**

python manage.py runserver 0.0.0.0:8000

Note: do not close the

# Stopping the Virtual Environment

When you're done working on your project virtual enviroment, you can deactivate the virtual environment:

virtualE\venv\scripts\deactivate

To reactivate the virtual environment later, navigate to your project directory and run the activate command again.

virtualE\venv\scripts\activate

**Remember to always activate your virtual environment before working on your Django project. This ensures that you're using the correct versions of Django and other dependencies specific to this project.**