**GE02 Django & Virtual Environments Documentation**

**Getting Started**

Here is what we have installed:

* Python (Version 3.11.5) [Can be checked by using py --version]
* Pip (Version 23.2.1) [Can be checked by using pip --version]
* Visual Studio Code
* Git & GitHub Repository

**Installing Django & Setting up a Virtual Environment**

*The commands below were completed on Windows and may be different for Mac, use:* [*https://www.w3schools.com/django/django\_install\_django.php*](https://www.w3schools.com/django/django_install_django.php)

1. Create a personal portfolio folder inside a CS3300 folder.
2. Inside the terminal, make your current working directory the portfolio folder, using the cd command. (*cd path\_to\_portfolio\_folder*)
3. Create a virtual environment by using the command: *py -m venv djvenv.* You should see a new folder in your portfolio named djvenv. This is the virtual directory that you will be using.
4. You must now activate the virtual environment, using the command: *djvenv\Scripts\activate.bat.* After the virtual environment has been activated, you should see *(djvenv) C:\your\_path\_here>* in the command line. Your virtual environment is now in use.
5. To install Django, you must use the command: *py -m pip install Django.* You should see something like this:

*A screen shot of a computer

Description automatically generated*

Django has now been successfully installed within your virtual environment.

1. Use the following command to create a Django Project: *django-admin startproject django\_project.* You should see a new folder titled “django\_project” within your portfolio folder.
2. Next you will want to reorganize your folders. Opening GIT Bash inside of the portfolio folder, preform the following commands:

A screen shot of a computer

Description automatically generated

These commands provided in the guided exploration will rearrange your folders in a more convenient way.

1. Back in the regular terminal, run the command: *py manage.py runserver.* You should see a short blurb mentioning unapplied migrations. Next, use this link: <http://localhost:8000/>

You should see a screen that looks like this:

A rocket with text on it

Description automatically generated with medium confidence

This screen means Django is working.

1. Open another terminal. The previous terminal will be used to start and stop the server, and this new terminal will be used to run commands. A third GIT bash terminal can be used for Git commands. Run the command: *pip freeze > requirements.txt.* This will create a file in the portfolio folder listing the required versions of different programs.

Django and your Virtual Environment should now be completely set up.

**Remember:** Use steps 2 & 4 in order to activate the Virtual Environment.

**Visual Studio Code**

In Visual Studio Code, open your portfolio folder. This is your project.

**Creating a Private Repository**

1. Create a new repository using the path to your portfolio. This is your own personal private repository.
2. Add a “.gitignore” file by running GIT Bash and using the command: *touch .gitignore*. The files/directories specified in this file will be ignored when committing changes in your GitHub. Here is an example of what a “.gitignore” file can look like: <https://github.com/github/gitignore/blob/main/Python.gitignore>

You will want to add “/djvenv” to the file, so it will ignore the folder holding your virtual environment.

1. I recommend installing GitHub Command Line (CLI), as it is synced with your GitHub and allows you to perform Git commands within your terminal.