Exercise 2.1 Task

1. Write two to three sentences on why Django is so popular among web developers.
   Django is so popular amongst developers for a few main reasons. To start, it offers such a wide range of security features out of the box. Additionally, due to the pre-built components and functionalities offered the development process runs quickly and more efficiently amongst development teams. Finally, Django has a large community of contributors which has led to easily accessible documentation and community discussions for any issues you may run into.

2. After some research, list five large companies that use Django. Specify what the company's product or service is and what they use Django for.
   - Spotify - music streaming platform
     - Spotify uses Django in their web player. More specifically, they make use of Django for handling user accounts, playlists, liked songs, and other features that users interact with directly on their profile.
   - Pinterest - social media platform
     - Pinterest uses Django for their web-based version of the app. It is being used to serve dynamic pages across their ever-scaling platform.
   - Dropbox - cloud storage service
     - Dropbox uses Django to help manage the large/varying amounts of data that a user may interact with on their account.
   - Udemy - online learning platform
     - Udemy uses Django to ensure the scalability of their site. This includes managing the vast bank of course material, and user interaction with that material.
   - Mozilla - developer of web technologies (most notably Firefox)
     - Mozilla uses Django to manage the large amounts of content being served, and the large user base interacting with that content.

3. For each of the following scenarios, explain if you would use Django (and why or why not):
   - You need to develop a web application with multiple users.
     - Yes - Django's out of the box features offer a wide range of support directly related to user management. This includes authentication, registration, and user permissions.

   - You need fast deployment and the ability to make changes as you proceed.
     - Yes - Django follows a "batteries-included" mindset which speeds up the development process. It is also fairly flexible when it comes to making changes throughout development.

- You need to build a very basic application, which doesn't require any database access or file operations.
    - No - Implementing Django in this scenario would be excessive and end up slowing down the development process due to its complexity, and slow down the application itself.

- You want to build an application from scratch and want a lot of control over how it works.
    - No - Django does things the way Django does things. This can be very helpful in situations where you can accept not having full control over your application. However, in this scenario Django would not be the best option as you'd lose a good amount of control during the development of your app.

- You're about to start working on a big project and are afraid of getting stuck and needing additional support.
    - Yes - Due to Django's popularity there is a vast community built around the framework. There are constant contributions being made and the documentation remains up-to-date. Additionally, there are many discussions happening online about topics pertaining to Django.

Exercise 2.1 Screenshots

1. Virtual Environment Set Up

```
EXPLORER                    ...

∨ PYTHON FOR WEB DEVE...
  > Achievement1
  ∨ Achievement2/2.1/a...
    > bin
    > include
    > lib
    ◆ .gitignore
    ⚙ pyvenv.cfg
    ⓘ README.md
```

```
PROBLEMS   OUTPUT   TERMINAL   PORTS

● jfrancis@jfrancis 2.1 % python3 -m venv achievement2-practice
● jfrancis@jfrancis 2.1 % cd achievement2-practice
● jfrancis@jfrancis achievement2-practice % source bin/activate
○ (achievement2-practice) jfrancis@jfrancis achievement2-practice % ▮
```

2. Django Install

```
● (achievement2-practice) jfrancis@jfrancis achievement2-practice % pip install django
Collecting django
  Using cached Django-5.1.5-py3-none-any.whl.metadata (4.2 kB)
Collecting asgiref<4,>=3.8.1 (from django)
  Using cached asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
Using cached Django-5.1.5-py3-none-any.whl (8.3 MB)
Using cached asgiref-3.8.1-py3-none-any.whl (23 kB)
Using cached sqlparse-0.5.3-py3-none-any.whl (44 kB)
Installing collected packages: sqlparse, asgiref, django
Successfully installed asgiref-3.8.1 django-5.1.5 sqlparse-0.5.3
● (achievement2-practice) jfrancis@jfrancis achievement2-practice % django-admin --version
  5.1.5
```