EXERCISE 2.1: GETTING STARTED WITH DJANGO

1. Suppose you're a web developer in a company and need to decide if you'll use vanilla (plain) Python for a project, or a framework like Django instead. What are the advantages and drawbacks of each?

Vanilla Python: Advantages: Lightweight, flexible, and relatively easy to learn. One of the main disadvantages of using vanilla Python is that it can be more time-consuming to develop projects as compared to using a framework.

Django: Advantages: Rapid development, built-in features and tools, scalable, and secure. One of the main disadvantages of using Django is that it can be more difficult to deploy projects as compared to using vanilla Python.

2. In your own words, what is the most significant advantage of Model View Template (MVT) architecture over Model View Controller (MVC) architecture?

The most significant advantage of MVT over MVC is that MVT separates the application logic into three interconnected components, making it easier to maintain and update the application. In MVT, the Template is responsible for rendering the user interface, while in MVC, the View handles both rendering and application logic.

3.   Now that you've had an introduction to the Django framework, write down three goals you have for yourself and your learning process during this Achievement. You can reflect on the following questions if it helps:

● *What do you want to learn about Django?*

● *What do you want to get out of this Achievement?*

● *Where or what do you see yourself working on after you complete this Achievement?*

I want to learn Django fundamentals such as architecture, models, views, templates, databases and I'd like to be confident enough to work with different features and tools of Django. In the future I want to apply my knowledge by building a real-world project using Django.