

## Exercise 2.1: Getting Started with Django

### Learning Goals

- Explain MVT architecture and compare it with MVC
- Summarize Django's benefits and drawbacks
- Install and get started with Django

### Reflection Questions

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?

The advantages of using Django are:

- Django is implemented in python, making it easy to support in the backend with compromising the front end
- Django allows for fast, easy development because of it's MVT architecture
- Django's architecture allows for fast handling of network transmission and content delivery
- Django's use of DRY principles will keep code non-repetitive
- Django allows for easy setup of CDNs, making it easy to manage and support content streaming
- Django makes it very simple to add new hardware and components, making it very easy to scale a project
- Django is built with security in mind, keeping data safe.
- Django is open source with a huge community of contributors making it easy to find support.

The disadvantages of Django are:

- Django is highly structured, meaning you have to do things a certain way. If your project requires a lot of control over how things are done Django is not the right choice.
- Django isn't suited for simple projects. If you don't need to access a database, manage files, or require many of Django's built in features it's probably not the right choice for your project.

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 architecture over MVC architecture for a developer is not needing to worry about writing code in order to fetch data from the database and mapping it to the URL. MVT allows a developer to just specify what elements they want to present to the user and then the framework does all of the heavy lifting of preparing these elements and sending them. This allows the developer to spend more time focusing on the unique code needed for their application.

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?

I want to learn how best to leverage Django, and take advantage of its built in features to streamline the development of future apps.

- What do you want to get out of this Achievement?

I want to get a stronger understanding of Django, and how it integrates with Python. I want to come out of this Achievement feeling as comfortable working with Django as I now feel working with Python after Achievement 1.

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

After I complete this Achievement I see myself taking the time to brainstorm a personal project that allows me to better show off my understanding of Python, Django, and SQL. I want to create an application from scratch that can be a strong portfolio piece. Perhaps, this could be centered around predicting the winners of NFL games as this is an interest I am passionate about outside of coding.