

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?

Using vanilla Python gives you full control over how you structure your web application, offering flexibility and minimal overhead. However, it requires more effort to set up basic web functionalities like routing, security, and database management.

On the other hand, a framework like Django provides built-in features such as authentication, ORM, and admin interface, speeding up development and enforcing best practices. The drawback is that Django's structure can be restrictive, and it may introduce unnecessary complexity for small or highly customized projects.

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 simplifies development by handling the controller logic automatically, allowing developers to focus more on defining models, views, and templates. In MVT, the framework (like Django) manages much of the controller's functionality, reducing boilerplate code and making development more efficient.

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?
 - How to build a Python app with a robust front-end framework that Django is and generally how Django implements the MVT architecture, I'd like to understand how this architecture works with this framework's example.
- What do you want to get out of this Achievement?
 - Build an app with a user-friendly interface in Python and Django to also add to my portfolio and enhance my skills stack with more advanced knowledge of Python.
- Where or what do you see yourself working on after you complete this Achievement?

- I still need to narrow down my goals after I finish this achievement. For now, I am in a mode of exploring full-stack and Python development, want to learn as much as possible, and build real apps. Once I finish this course, I'll clarify my goal which is currently rather broad - I'd like to focus on front-end development mainly but I am open to full-stack as well.