

Exercise 2.4: Django Views and Templates

Learning Goals

- Summarize the process of creating views, templates, and URLs
- Explain how the “V” and “T” parts of MVT architecture work
- Create a frontend page for your web application

Reflection Questions

1. Do some research on Django views. In your own words, use an example to explain how Django views work.
 - Django views handle incoming web requests and define the logic for generating appropriate HTTP responses. They are linked to specific URLs, allowing for effective request routing in a web application.
2. Imagine you're working on a Django web development project, and you anticipate that you'll have to reuse lots of code in various parts of the project. In this scenario, will you use Django function-based views or class-based views, and why?
 - If you anticipate reusing code in different parts of your Django project, it's recommended to use class-based views (CBVs). CBVs provide a structured and modular approach, making it easier to organize and maintain code, fostering better code reuse compared to function-based views.
3. Read Django's documentation on the Django template language and make some notes on its basics.
 - Django Template Language (DTL) enables developers to integrate dynamic content into web pages using a Python-like syntax within HTML templates. DTL basics include using curly braces for variable insertion, `{% tags %}` for control flow, and `{{variables}}` for outputting content, facilitating the dynamic rendering of data in Django templates.