

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.

A view in Django is the logic acts as a function or a method of a Python class that accepts a request, runs code and returns a response. The response can be as basic as any static information like an HTML page or more complex like a Netflix-like search where a class accepts a request from the user, interacts with the database using the user input and determines what will be returned as a response.

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?

The best view-type to use in this scenario would be the class-based views because they reduce the effort of duplicating or rewriting code because they've already been defined as classes. They are best used when you need generic view reused across multiple apps, like `ListView()`.

3. Read Django's documentation on the Django template language and make some notes on its basics.

A template is a text file that can generate any tex-based format (HTML, XML, CSV, etc) and contains variable which are replaced with values when the template is evaluated and tags which control the logic of the template.