

2.4 Reflection Questions

1. Do some research on Django views. In your own words, use an example to explain how Django views work.

Django Views are the HTML that will be rendered for each page in the application. Each app that represents a different view can be mapped to a different path on the main URL.

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?

We would most likely want class based views (CBT) in this instance. There are already some built in generic views like ListView, DetailView, CreateView and UpdateView. These prebuilt views could be reused to cut down on code duplication.

3. Read Django's documentation on the [Django template language](#) and make some notes on its basics.

Variables: Use `{{ variable }}` to display data.

Tags: Control logic with `{% tag %}`, e.g., `{% if %}` and `{% for %}`.

Filters: Modify output with `{{ value|filter }}`.

Static Files: Load with `{% load static %}`.

URLs: Link views with `{% url 'view_name' %}`.