

Exercise 2.2: Django Project Set Up

Learning Goals

- Describe the basic structure of a Django project
- Create a Django project and run it locally
- Create a superuser for a Django web application

Reflection Questions

- Suppose you're in an interview. The interviewer gives you their company's website as an example, asking you to convert the website and its different parts into Django terms. How would you proceed? For this question, you can think about your dream company and look at their website for reference.

(Hint: In the Exercise, you saw the example of the CareerFoundry website in the Project and Apps section.)

- Identify the main features of the website and map them to Django apps—each app representing a key functionality, such as user authentication, blog, or product catalog.
 - The overall website would be the project, managing settings, URLs, and global configurations. Each page or section of the website would correspond to views in Django, and the visual elements would be handled through templates. Finally, any dynamic data, like user profiles or product listings, would be managed by models in the respective apps.
- In your own words, describe the steps you would take to deploy a basic Django application locally on your system.

To deploy a basic Django application locally, follow these steps:

1. **Set Up the Environment:** Install Python and Django on your system. You can create a virtual environment using `python -m venv venv` and activate it to keep dependencies isolated.
2. **Create a Django Project:** Run `django-admin startproject project_name` to create a new Django project.
3. **Create an App:** Inside your project directory, create an app with `python manage.py startapp app_name`.
4. **Configure the Project:** Update `settings.py` to configure installed apps, database settings, and other configurations.
5. **Migrate the Database:** Run `python manage.py migrate` to apply database migrations.
6. **Run the Development Server:** Use `python manage.py runserver` to start the local development server.
7. **Access the Application:** Open a web browser and navigate to `http://127.0.0.1:8000/` to see your Django application running locally.
8. **Test and Develop:** You can now add views, templates, and models to develop your application further.

- Do some research about the Django admin site and write down how you'd use it during your web application development.

The Django admin site is a built-in interface that allows developers and site administrators to manage and interact with the application's data models.

By registering models in the `admin.py` file of an app, Django automatically generates a user-friendly admin interface for creating, reading, updating, and deleting entries in the database. It also provides features like search, filtering, and user management out of the box. The admin site is accessible by default at `/admin/`.