

Exercise 2.2 Learning Journal

Reflection Questions

1. **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.**

Dream Company Website Features:

1. Homepage: Showcases the company's latest products and news.
2. Products Page: Lists all products with details and filters.
3. Product Detail Page: Shows detailed information about a specific product.
4. About Us Page: Information about the company and team.
5. Contact Page: Allows users to contact the company and submit inquiries.
6. User Authentication: Users can sign up, log in, and manage their profiles.
7. Admin Dashboard: For managing products, blog posts, and user inquiries.

The whole website is the "Project". Converting the website into Django terms involves mapping the different components of the site to Django's architectural elements, such as models, views, templates, and URLs.

2. **In your own words, describe the steps you would take to deploy a basic Django application locally on your system.**

Make sure you have Python installed. Use pip to install Django. Use the Django command-line tool to create a new project. Django uses migrations to handle database schema changes. Apply the initial migrations. Create a superuser account to access the Django admin interface. Start the development server.

3. **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 powerful feature that can greatly enhance your web application development process. It provides a built-in interface for managing and interacting with the application's data models, allowing for quick and efficient administrative tasks. Creating superusers is one of the features. You can register and manage models from the admin interface. You can customize the forms used in the admin interface. If you have related models, you can use inlines to manage them within the same admin page. Define custom actions to perform bulk operations on selected items. Enhance the usability of the admin interface by adding filters and search functionality. Regularly use the admin interface to test model changes, review data, and ensure everything is working as expected. As your application evolves, continuously refine your admin configurations to match the needs of your project.