Exercise 2.6: User Authentication in Django

Learning Goals

- Create authentication for your web application
- Use GET and POST methods
- Password protect your web application's views

Reflection Questions

1. In your own words, write down the importance of incorporating authentication into an application. You can take an example application to explain your answer.

   It's important to incorporation authentication into your applications in case there is sensitive data to protect or if there's certain views that are restricted to certain users. By using a method of authentication, you can control who has access to what information and how they interact with it. If a user of the recipe app wants to add a recipe to their account, you wouldn't want someone to go behind them and be able to delete the recipe.

2. In your own words, explain the steps you should take to create a login for your Django web application.

   To create a login for your Django application, you need to create a view in which you import authenticate, login, and AuthentciationForm. Using these imports, you can then write the logic for what happens when the login button is clicked by the user. Then you create the template that provides the form to the user to enter their information. Once you have the view and template created, you can add the view to the urlpatterns inside the project urls.py file.

3. Look up the following three Django functions on Django's official documentation and/or other trusted sources and write a brief description of each.

Function	Description
authenticate()	Verifies a set of credentials that it takes as keyword arguments to check against the backend and returns a User object.
redirect()	Redirects user to a URL; arguments passed to the function can be a model, a view name, or an absolute / relative URL
include()	This function can be used to add the urls from app directories to the main url.py file