

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

- ❖ In your own words, write down the importance of incorporating authentication into an application. You can take an example application to explain your answer.
 - Incorporating authentication into an application is important to protecting sensitive user information as well as allowing user-specific features and content to cater to the individual user. Take the YouTube website for example. Anyone can access the site and view publicly published videos, but you need to sign in (as an authenticated user) before you can access information and content specific to you - like your watch history, video recommendations, and your own published content along with options to manage them. This layer of authentication protects other people from seeing your watch history (privacy) and accessing your content without permission.
- ❖ In your own words, explain the steps you should take to create a login for your Django web application.
 - To create a login for my Django web application, I would first create a view for taking care of the login function. This view should load the login template with the authentication form (requesting input from user), and this template would display the login form from the view, including the Django security token and submit button. Then I would register the URL to the project (in the root urls.py file). I would also need to ensure the proper authentication libraries, functions, and Form are imported, and that the TEMPLATES list variable is properly updated in the settings.py file (to let Django know where to look for the templates).
- ❖ 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 against each authentication backend, and returns a User object if the credentials are valid
redirect()	Takes an argument (could be a model, a view name, or an absolute/relative URL) and returns by default a temporary redirect to that URL
include()	A function that takes a full Python import path to another URLconf module that should be "included" in this place; imports & renders a template, allowing inclusion of other templates within a template