

EXERCISE 2.6: USER AUTHENTICATION IN DJANGO

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.

Incorporating authentication into an application is crucial to ensure that only authorized users can access sensitive data and features. For example, consider an online banking application. Without authentication, anyone could access and manipulate users' account info, leading to security breaches and financial losses. By incorporating authentication, the application can verify users' identities and grant access only to authorized individuals, protecting sensitive data and preventing unauthorized actions.

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

- Install the `django.contrib.auth` module
- Create a User model (e.g., using `AbstractBaseUser`)
- Create a login form and view
- Use the `login()` function to authenticate users
- Redirect authenticated users to a protected page
- Use decorators (e.g., `@login_required`) to restrict access to protected pages

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
<code>authenticate()</code>	Verifies a user's credentials (e.g., username and password) and returns a User object if authenticated.
<code>redirect()</code>	Redirects the user to a different URL, often used after a successful login or form submission.
<code>include()</code>	Django function used add urls from apps directory to the main <code>urls.py</code> file in the project directory.