Description Of Project:

Step 1: Generate the Maven project using Spring Initializer and import it into Eclipse.

Use Spring Initializer to generate a Maven project.

Download the generated zip file and import it into Eclipse.

Step 2: Add dependencies and configure application properties.

Modify the pom.xml file to include the necessary dependencies for your project.

Edit the application.properties file to configure the database connection.

Step 3: Create homepage and functionality for user interaction.

Create a homepage file to serve as the entry point of your application.

Add functionality to direct the user to the all users page, which displays the users in the database.

Implement a form for users to input their name, email, and password to add themselves to the database.

Step 4: Create controller class for mapping URLs.

Create a controller class to handle mapping for the homepage, login page, and all users page.

Define methods in the controller class to handle different URL requests.

Step 5: Create User class and initialize it.

Create a User class with attributes like id, name, email, and password.

Implement getters and setters for these attributes.

Initialize a User object with the necessary values.

Step 6: Create UserRepository to interact with the database.

Create a UserRepository class that extends CrudRepository, allowing you to make changes to the database.

Use UserRepository to find and modify data in the database.

Step 7: Create UserService to handle database operations.

Create a UserService class that uses UserRepository to retrieve data from the database.

Implement methods in UserService to perform operations like finding users in the database.

Step 8: Apply CSS changes to the web pages.

Customize the appearance of your web pages using CSS.

Make changes to center text and add a background, for example.

Step 9: Upload project To github

Testing:

- Write tests to ensure the correctness and functionality of your code.
- In AuthenticationApplicationTests, verify that the context loads and the database has users.
- In AuthenticationTests, test that user information is added correctly and not null.
- In AuthenticationWebTests, check the functionality of the web pages.
- In EntityTests, test getters, setters, constructors, and toString methods of the User class.

Algorithm:

- Generate Maven project using Spring Initializer and import it into Eclipse.
- Modify the pom.xml file to include dependencies and configure application properties.
- Create homepage and implement user interaction functionality.

- Create a controller class for mapping URLs and handle different requests.
- Implement the User class with necessary attributes, getters, and setters.
- Create UserRepository to interact with the database and UserService to handle database operations.
- Apply CSS changes to customize the appearance of web pages.
- Upload the project to GitHub for version control.
- Write tests to ensure the correctness and functionality of the code.
- Execute the tests to validate the behavior of the application.

Github Repo: https://github.com/Aryan9605garg/AryanPracticeRepo