Create bookstore folder inside webapps

Inside bookstore keep

**LoginServlet.java**

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.io.IOException;

import java.sql.\*;

public class LoginServlet extends HttpServlet {

    private static final String URL = "jdbc:mysql://localhost:3306/bookstore";  // MySQL database URL

    private static final String DB\_USER = "root";  // Database username

    private static final String DB\_PASSWORD = "password";  // Database password (replace with actual)

    // JDBC variables for opening, closing, and managing the database connection

    private Connection connection;

    private PreparedStatement preparedStatement;

    private ResultSet resultSet;

    @Override

    public void init() throws ServletException {

        try {

            // Load MySQL JDBC driver

            Class.forName("com.mysql.cj.jdbc.Driver");

            // Establish connection to the database

            connection = DriverManager.getConnection(URL, DB\_USER, DB\_PASSWORD);

        } catch (Exception e) {

            throw new ServletException("Database connection error", e);

        }

    }

    @Override

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        String username = request.getParameter("username");

        String password = request.getParameter("password");

        // Authenticate user by calling authenticate method

        User user = authenticate(username, password);

        if (user != null) {

            // Store user in session

            HttpSession session = request.getSession();

            session.setAttribute("username", user.getUsername());  // Save username in session

            response.sendRedirect("home.jsp");   // Redirect to home.jsp after successful login

        } else {

            response.sendRedirect("login.jsp?error=Invalid credentials");  // Redirect to login.jsp with error message

        }

    }

    // Authenticate method to check if the username and password exist in the database

    private User authenticate(String username, String password) {

        User user = null;

        // Query to retrieve user data

        String sql = "SELECT \* FROM users WHERE username = ? AND password = ?";

        try {

            preparedStatement = connection.prepareStatement(sql);

            preparedStatement.setString(1, username);

            preparedStatement.setString(2, password);

            resultSet = preparedStatement.executeQuery();

            if (resultSet.next()) {

                // User found, create a User object and set the details

                user = new User();

                user.setUsername(resultSet.getString("username"));

                user.setPassword(resultSet.getString("password"));

                // You can retrieve other user data like email or full name if needed

            }

        } catch (SQLException e) {

            e.printStackTrace();

        }

        return user;

    }

    @Override

    public void destroy() {

        try {

            if (resultSet != null) resultSet.close();

            if (preparedStatement != null) preparedStatement.close();

            if (connection != null) connection.close();

        } catch (SQLException e) {

            e.printStackTrace();

        }

    }

}

**Login.jsp:**

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

    <title>Login</title>

</head>

<body>

    <h2>Login Page</h2>

    <form action="LoginServlet" method="post">

        <label for="username">Username:</label>

        <input type="text" name="username" id="username" required><br><br>

        <label for="password">Password:</label>

        <input type="password" name="password" id="password" required><br><br>

        <input type="submit" value="Login">

    </form>

    <!-- Display error message if login fails -->

    <c:if test="${not empty param.error}">

        <p style="color: red;">${param.error}</p>

    </c:if>

</body>

</html>

**Home.jsp:**

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

    <title>Welcome</title>

</head>

<body>

    <h2>Welcome, ${sessionScope.username}!</h2>

    <p>You have successfully logged in.</p>

    <a href="logout.jsp">Logout</a>

</body>

</html>

**Logout.jsp**

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<%@ page import="javax.servlet.http.HttpSession"%>

<%

    HttpSession session = request.getSession();

    session.invalidate(); // Invalidates the session

    response.sendRedirect("login.jsp"); // Redirect to login page after logout

%>

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<%@ page import="javax.servlet.http.HttpSession"%>

<%

    HttpSession session = request.getSession();

    session.invalidate(); // Invalidates the session

    response.sendRedirect("login.jsp"); // Redirect to login page after logout

%>

**User.java**:

public class User {

    private String username;

    private String password;  // You can add other fields as needed, like email, name, etc.

    // Getter and setter methods for username and password

    public String getUsername() {

        return username;

    }

    public void setUsername(String username) {

        this.username = username;

    }

    public String getPassword() {

        return password;

    }

    public void setPassword(String password) {

        this.password = password;

    }

}

**Then keep belowin lib of WEB-INF by downloading from maven(you may use chatgpt for links)**

C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\webapps\bookstore\WEB-INF\lib\jstl-1.1.2.jar

C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\webapps\bookstore\WEB-INF\lib\jstl-1.2.jar

C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\webapps\bookstore\WEB-INF\lib\mysql-connector-j-9.1.0.jar

**Compile with below:**

C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\webapps\bookstore>javac -cp "C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\lib\servlet-api.jar" -d WEB-INF\classes LoginServlet.java User.java

C:\Users\Alekhya\Downloads\apache-tomcat-9.0.97\apache-tomcat-9.0.97\webapps\bookstore>

**Web.xml**

<web-app xmlns="http://java.sun.com/xml/ns/javaee"

         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

         xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

                             http://java.sun.com/xml/ns/javaee/web-app\_4\_0.xsd"

         version="4.0">

    <!-- Servlet configuration for LoginServlet -->

    <servlet>

        <servlet-name>LoginServlet</servlet-name>

        <servlet-class>LoginServlet</servlet-class>

    </servlet>

    <!-- Servlet mapping for LoginServlet -->

    <servlet-mapping>

        <servlet-name>LoginServlet</servlet-name>

        <url-pattern>/LoginServlet</url-pattern>

    </servlet-mapping>

    <!-- Welcome file configuration -->

    <welcome-file-list>

        <welcome-file>login.jsp</welcome-file>

    </welcome-file-list>

</web-app>

Open xampp and run apache and sql and after that click on admin of sql and create new database in below way

CREATE DATABASE bookstore;

USE bookstore;

CREATE TABLE users (

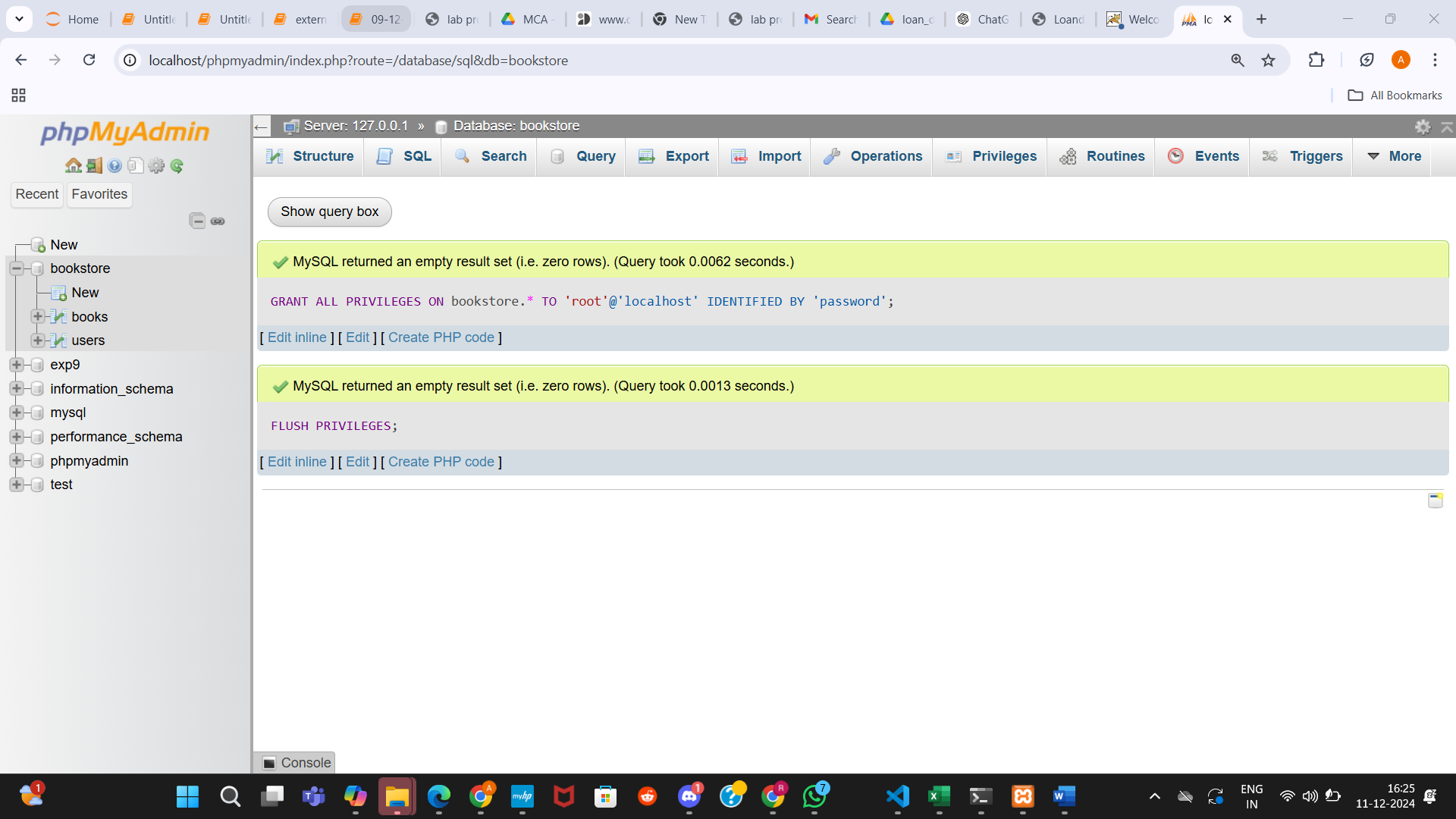
id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL,

password VARCHAR(50) NOT NULL

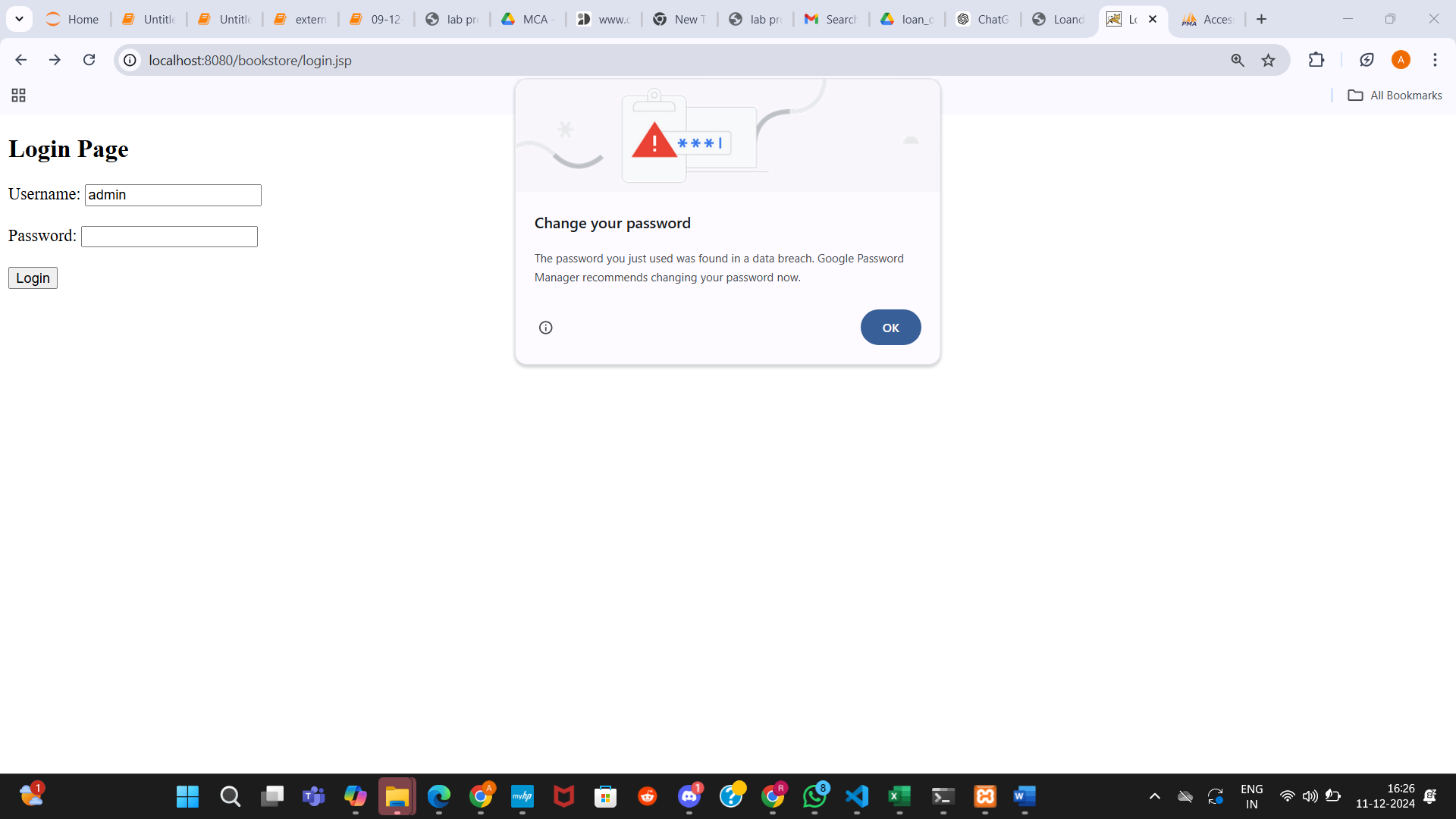
);

INSERT INTO users (username, password) VALUES ('admin', 'password');



Then on server of tomcat and go to

<http://localhost:8080/bookstore/login.jsp>



Here give username admin and password password

Upon clicking login you will see below

