ASSIGNMENT

Name: Anulekshmi A

Full Stack Development JAVA

24-07-2024

DAO (Data Access Object)

DAO stands for Data Access Object. DAO Design Pattern is used to separate the data persistence logic in a separate layer. This way, the service remains completely in dark about how the low-level operations to access the database is done. This is known as the principle of **Separation of Logic**.

Task1

Create a registration module with database connectivity to store data in a database

Task2

Create a login module with database connectivity to check authentication of user

Creating a Registration and Login System with Java (JDBC)

Step1: Create a New Java Project

1. Open NetBeans/Eclipes/Intelij.
2. Go to File -> New Project.
3. Select Java -> Java Application.
4. Click Next.
5. Enter the project name (e.g., RegistrationLogin) and location.
6. Click Finish.

Step2: Set Up the Project Structure

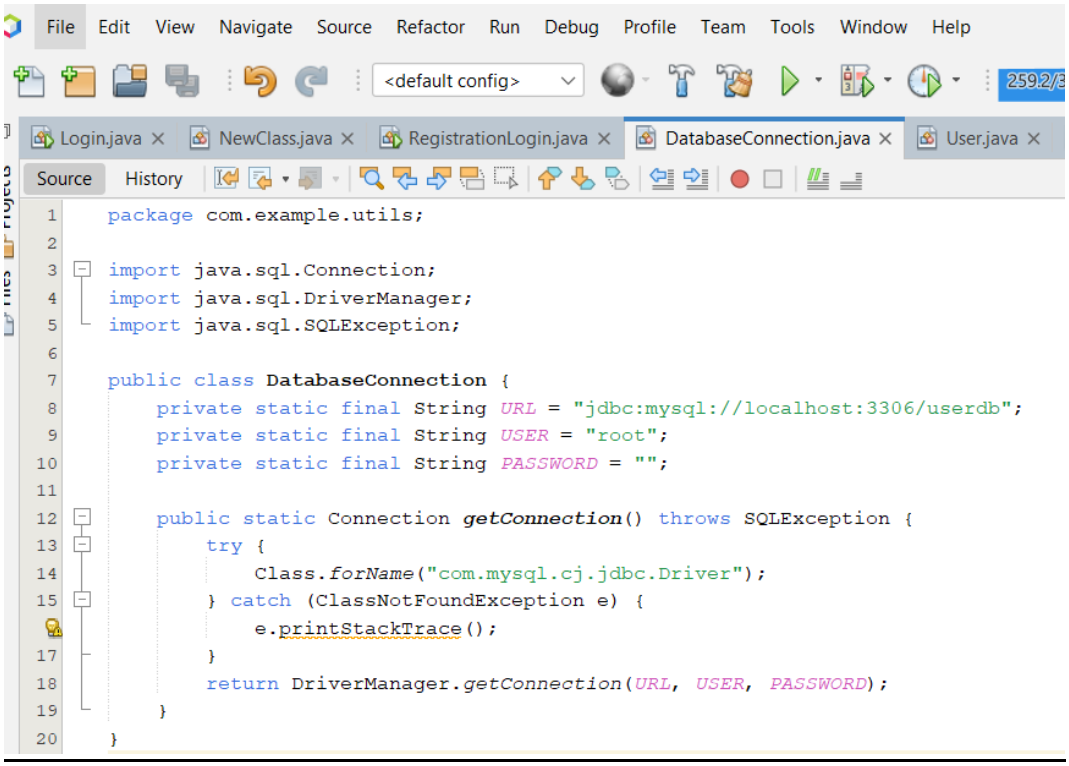
1. Right-click on the Source Packages directory in the Project Explorer.
2. Select New -> Java Package.

Step3: Add MySQL Connector/J to the Project

1. Right-click on the project name in the Project Explorer.
2. Select Properties.
3. Go to Libraries -> Compile.
4. Click Add JAR/Folder and select the downloaded MySQL Connector/J JAR file.
5. Click Open and then OK.

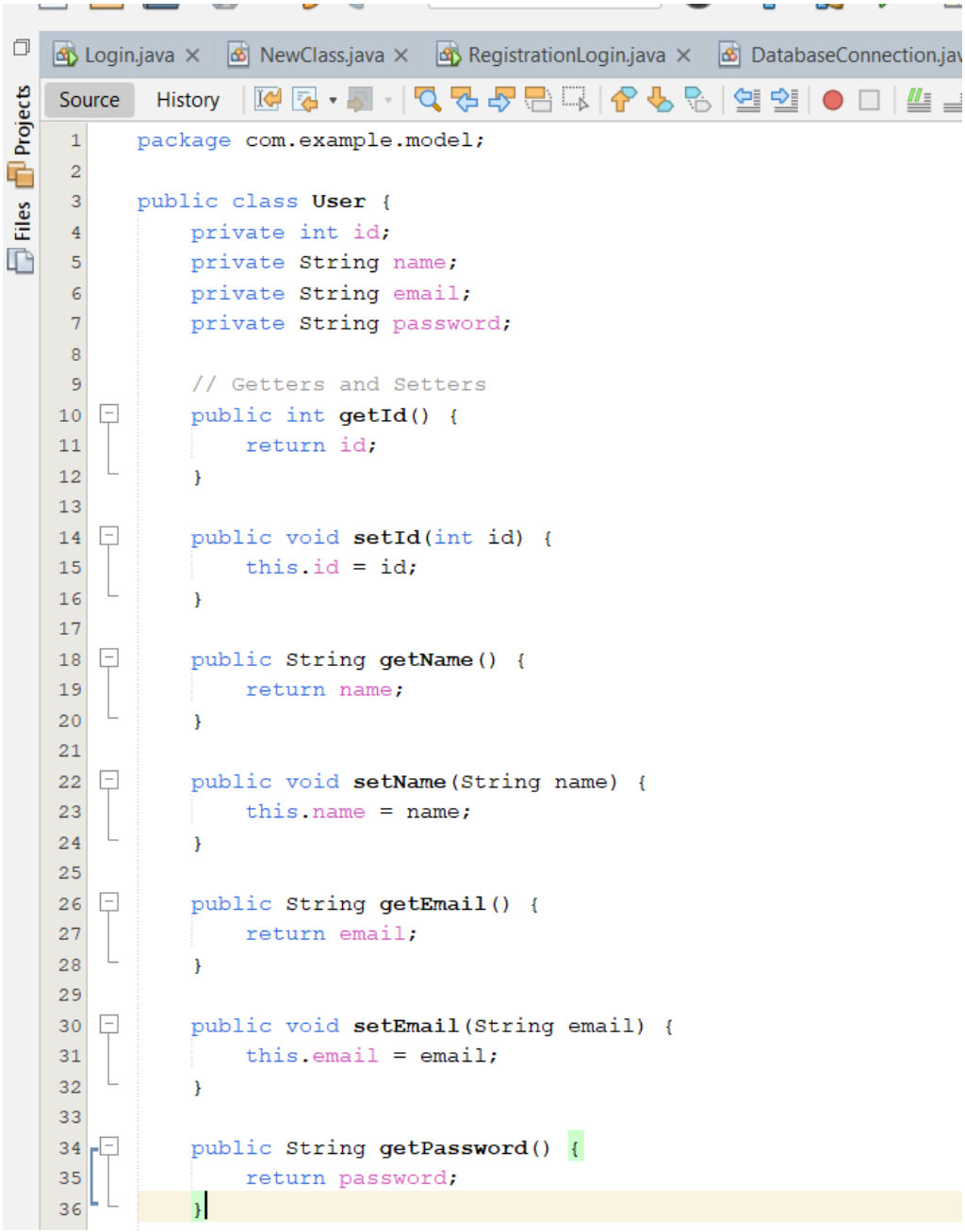
Step4: Create Java Classes

1. DatabaseConnection.java




```java
package com.example.utils;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {
    private static final String URL = "jdbc:mysql://localhost:3306/userdb";
    private static final String USER = "root";
    private static final String PASSWORD = "";

    public static Connection getConnection() throws SQLException {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
        return DriverManager.getConnection(URL, USER, PASSWORD);
    }
}
```

2.User.java



2. UserDao.java

```
package com.example.dao;

import com.example.model.User;

import com.example.utils.DatabaseConnection;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;
```

```
public class UserDao {
    public boolean registerUser(User user) {
        String query = "INSERT INTO users (name, email, password) VALUES (?, ?, ?)";
        try (Connection conn = DatabaseConnection.getConnection();
             PreparedStatement ps = conn.prepareStatement(query)) {
            ps.setString(1, user.getName());
            ps.setString(2, user.getEmail());
            ps.setString(3, user.getPassword());
            int result = ps.executeUpdate();
            return result > 0;
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return false;
    }
    public User loginUser(String email, String password) {
        String query = "SELECT * FROM users WHERE email = ? AND password = ?";
        try (Connection conn = DatabaseConnection.getConnection();
             PreparedStatement ps = conn.prepareStatement(query)) {
            ps.setString(1, email);
            ps.setString(2, password);
            try (ResultSet rs = ps.executeQuery()) {
                if (rs.next()) {
                    User user = new User();
                    user.setId(rs.getInt("id"));
                    user.setName(rs.getString("name"));
                    user.setEmail(rs.getString("email"));
                    user.setPassword(rs.getString("password"));
                    return user;
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
```

```
            return null;
        }
    }
```

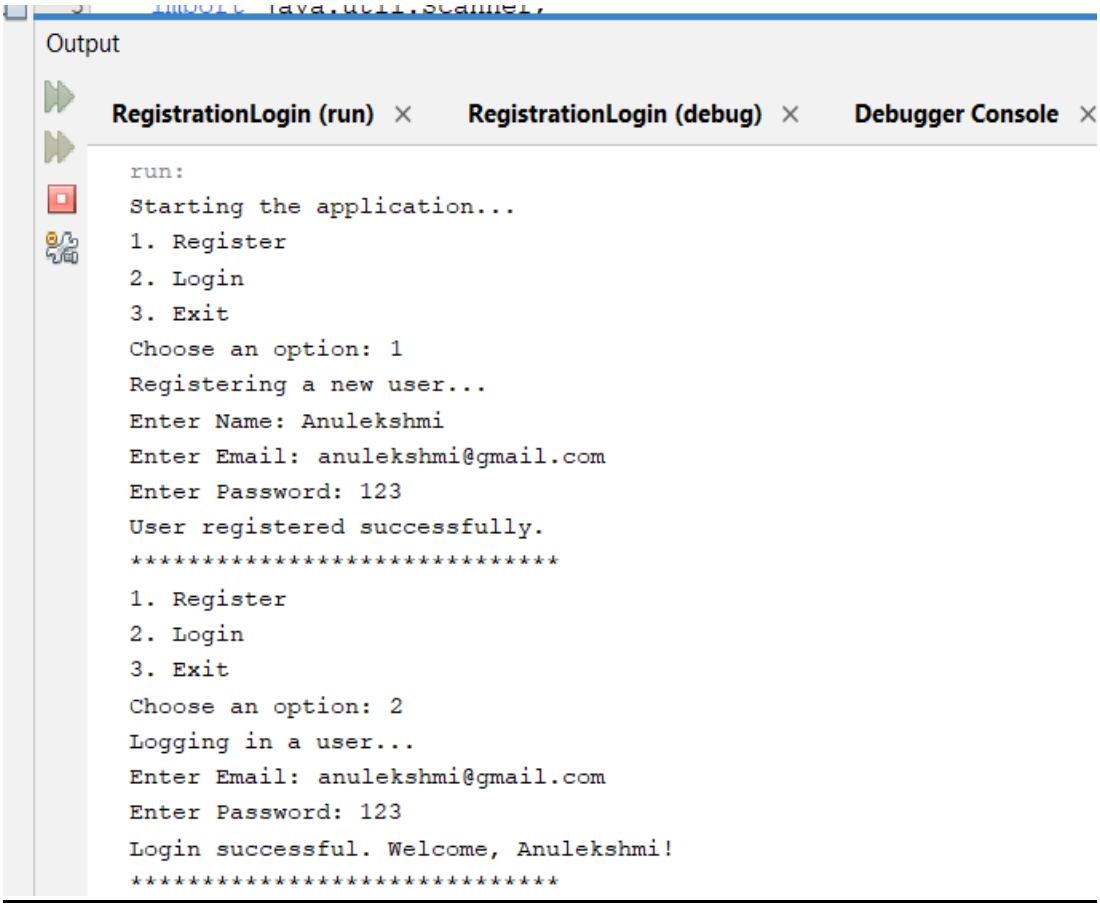
3. Main.java

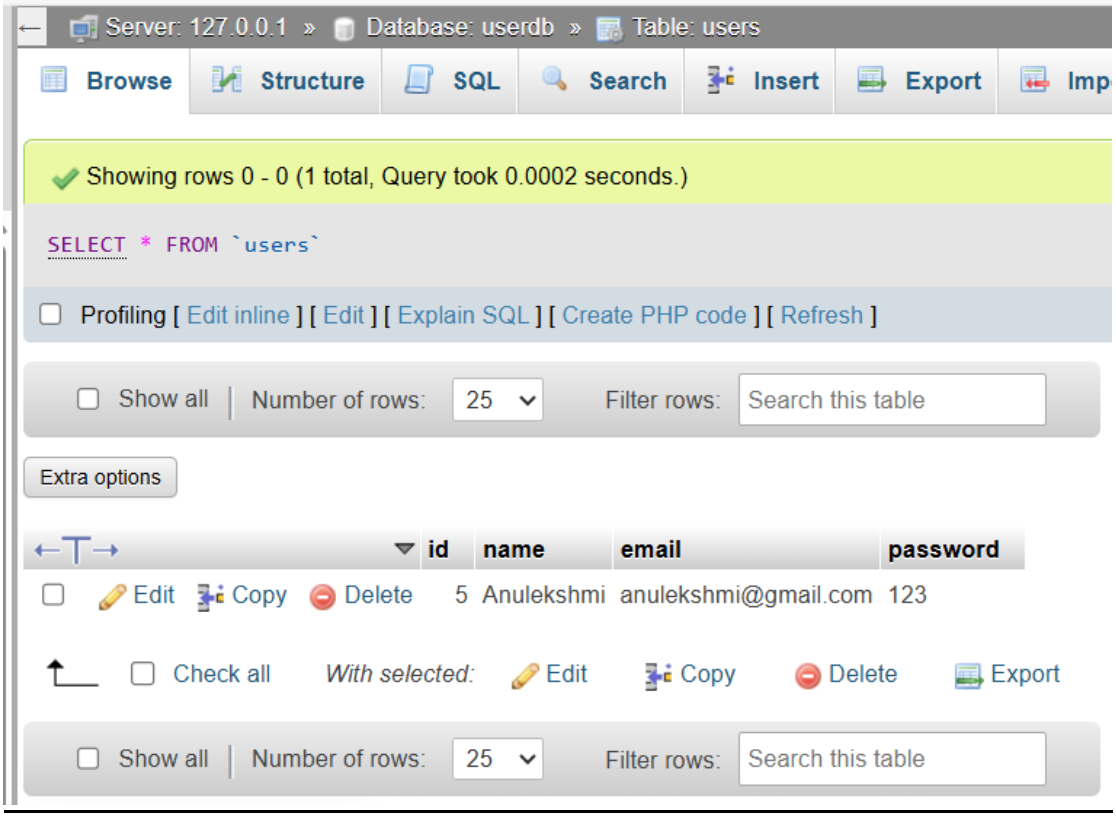
```
package com.example.main;
import com.example.dao.UserDao;
import com.example.model.User;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        System.out.println("Starting the application...");
        UserDao userDao = new UserDao();
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("1. Register");
            System.out.println("2. Login");
            System.out.println("3. Exit");
            System.out.print("Choose an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();  // Consume newline

            if (choice == 1) {
                System.out.println("Registering a new user...");
                System.out.print("Enter Name: ");
                String name = scanner.nextLine();
                System.out.print("Enter Email: ");
                String email = scanner.nextLine();
```

```
            System.out.print("Enter Password: ");
            String password = scanner.nextLine();

            User user = new User();
            user.setName(name);
            user.setEmail(email);
            user.setPassword(password);

            boolean isRegistered = userDao.registerUser(user);
            if (isRegistered) {
                System.out.println("User registered successfully.");
            } else {
                System.out.println("User registration failed.");
            }
        } else if (choice == 2) {
            System.out.println("Logging in a user...");
            System.out.print("Enter Email: ");
            String email = scanner.nextLine();
            System.out.print("Enter Password: ");
            String password = scanner.nextLine();

            User user = userDao.loginUser(email, password);
            if (user != null) {
                System.out.println("Login successful. Welcome, " +
user.getName() + "!");
            } else {
                System.out.println("Invalid email or password.");
            }
        } else if (choice == 3) {
```

```
            System.out.println("Exiting...");
            break;
        } else {
            System.out.println("Invalid option. Please try again.");
        }
    }
    scanner.close();
  }
}
```

OUTPUT


Output
RegistrationLogin (run)
RegistrationLogin (debug)
Debugger Console
run:
Starting the application...
1. Register
2. Login
3. Exit
Choose an option: 1
Registering a new user...
Enter Name: Anulekshmi
Enter Email: anulekshmi@gmail.com
Enter Password: 123
User registered successfully.
1. Register
2. Login
3. Exit
Choose an option: 2
Logging in a user...
Enter Email: anulekshmi@gmail.com
Enter Password: 123
Login successful. Welcome, Anulekshmi!


Server: 127.0.0.1 » Database: userdb » Table: users
Browse
Structure
SQL
Search
Insert
Export
Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)
SELECT * FROM `users`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all
Number of rows: 25
Filter rows: Search this table
Extra options
id
name
email
password
Edit
Copy
Delete
5
Anulekshmi
anulekshmi@gmail.com
123
Check all
With selected:
Edit
Copy
Delete
Export