



**UNIVERSITI MALAYSIA TERENGGANU**

---

**CSM3023 -PEMBANGUNAN APLIKASI BERASASKAN WEB**

**BANCHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING)  
WITH HONORS**

**LAB 2 – Servlet: Data Sharing and Database Management**

**SEMESTER 2 2023/2024**

---

**PREPARED FOR:**

**DR MOHAMAD NOR BIN HASSAN HASSAN**

**PREPARED BY:**

**NUR EZREENA SHUHADA BT EMRAN**

**S66467**

## Task 1 : Data Sharing in Servlet

### AccountServlet.java

```
import java.io.IOException;

import java.io.PrintWriter;

import java.util.HashMap;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * Nur Ezreena
 */

public class AccountServlet extends HttpServlet {

    HashMap<String,String[]> account = new HashMap();

    @Override

    public void init() throws ServletException{

        super.init();

        account.put("Ali", new String[]{"31/01/2019: 2000.00","28/02/2019: 3000.00"});

        account.put("Ahmad", new String[]{"31/01/2019: 100.00","28/02/2019: 5000.00"});

        account.put("Muthu", new String[]{"31/01/2019: 1000","28/02/2019: 2000"});
```

```
}
```

```
protected void processRequest(HttpServletRequest request, HttpServletResponse  
response)
```

```
throws ServletException, IOException {
```

```
response.setContentType("text/html;charset=UTF-8");
```

```
String userid_login = (String)request.getAttribute("userid");
```

```
try (PrintWriter out = response.getWriter()) {
```

```
    out.println("<!DOCTYPE html>");
```

```
    out.println("<html>");
```

```
    out.println("<head>");
```

```
    out.println("<title>Servlet AccountServlet</title>");
```

```
    out.println("</head>");
```

```
    out.println("<body>");
```

```
    if (account.get(userid_login) == null){
```

```
        out.println("<h1>Sorry, no information found!</h1>");
```

```
    }
```

```
    else{
```

```
        out.println("<h1>Account status for: " +userid_login+"</h1>");
```

```

        for(String tempAcc: account.get(userid_login)){

            out.println("<h2>" +tempAcc+ "</h2>");

        }

    }

    out.println("</body>");

    out.println("</html>");

}

}

```

## LoginServlet.Java

```

import jakarta.servlet.RequestDispatcher;

import jakarta.servlet.ServletContext;

import java.io.IOException;

import java.util.HashMap;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @Nur Ezreena
 */

public class LoginServlet extends HttpServlet {

    HashMap <String,String> users = new HashMap();

```

@Override

```
public void init() throws ServletException{
```

```
    super.init();
```

```
    users.put("Ali", "1234");
```

```
    users.put("Ahmad", "4567");
```

```
    users.put("Muthu", "8910");
```

```
}
```

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {
```

```
    response.setContentType("text/html;charset=UTF-8");
```

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

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

```
if(!username.equals(" ") && !password.equals(" "))
```

```
    && users.get(username).equals(password)){
```

```
    request.setAttribute("userid", username);
```

```
    ServletContext sc = getServletContext();
```

```
    RequestDispatcher rd = sc.getRequestDispatcher("/AccountServlet");
```

```
    rd.forward(request,response);
```

```
}else{
```

```
    //avoid direct access to the servlet
```

```
    RequestDispatcher rd = request.getRequestDispatcher("/login.html");
```

```
        rd.forward(request,response);

    }

}
```

## Login.html

```
<html>

<head>

    <title>Login Page</title>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <style>

        body{

            background-color: black;

            text-align: left;

            color:white;

            font-family: Arial,Helvetica,sans-serif;

        }

    </style>

</head>

<body>

    <h1>WELCOME TO CSF3107</h1>

    <p>Please insert your username and password</p>

    <form name="login" id="login" action="LoginServlet" method="POST" autocomplete="off">
```

```
Username:<input name="txtUsername" type="text"><br>

Password:<input name="txtPassword" type="text"><br>

<br>

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

<input name="txtReset" value="Reset" type="reset"><br><!-- comment -->


</form>

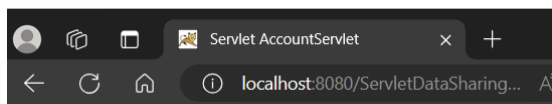
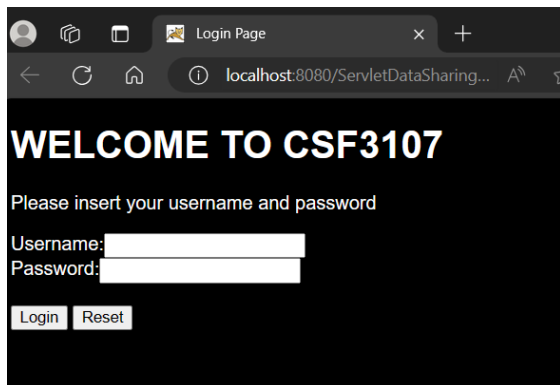
<p><br><!-- comment -->

</p><!-- comment -->

</body>

</html>
```

Output:

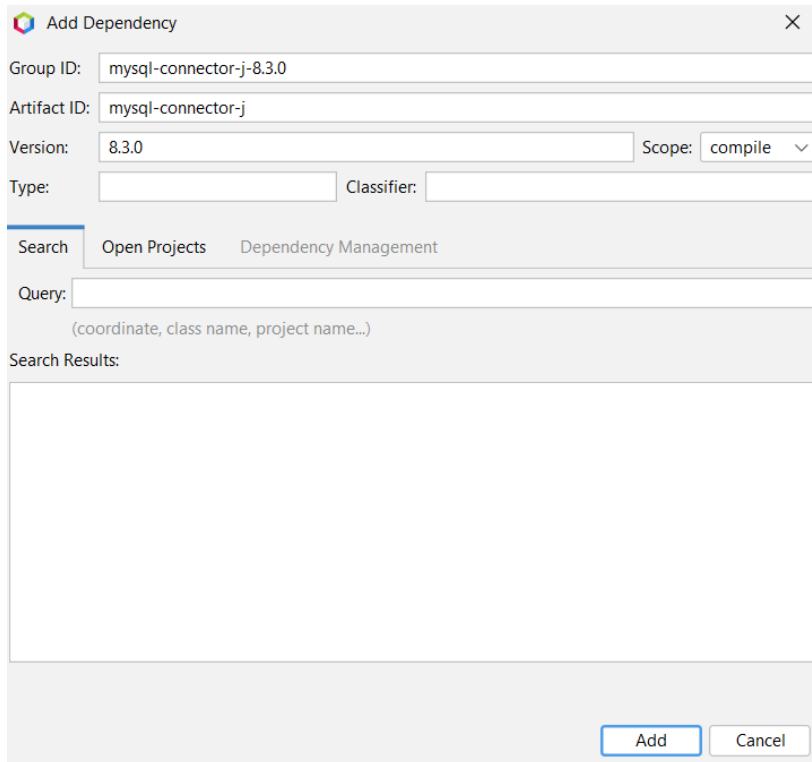


**Account status for: Ahmad**

**31/01/2019: 100.00**

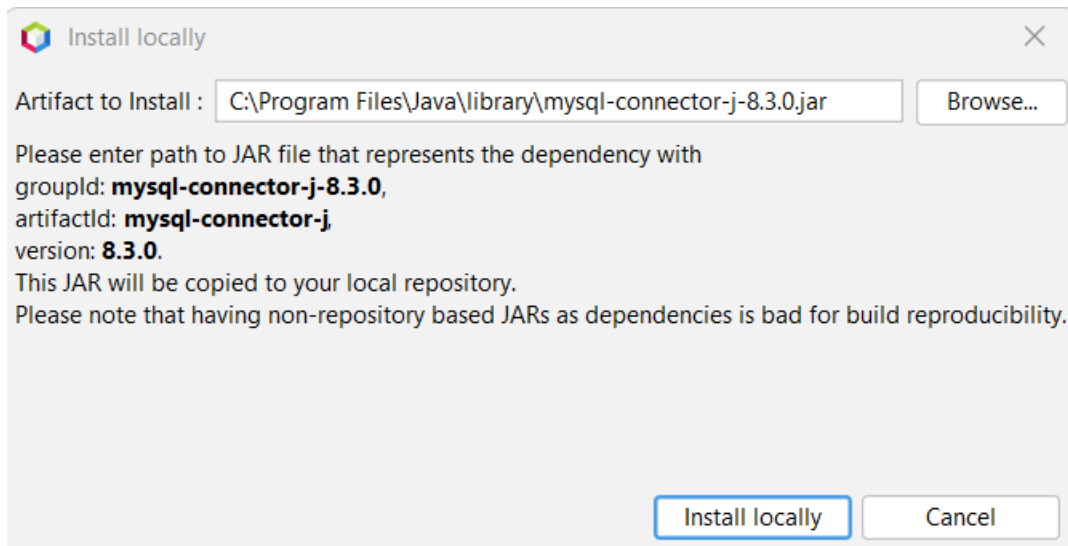
**28/02/2019: 5000.00**

## Task 2 : Creating A Table in MySQL Database



The "Add Dependency" dialog box is used to specify a dependency for a project. It contains the following fields and options:

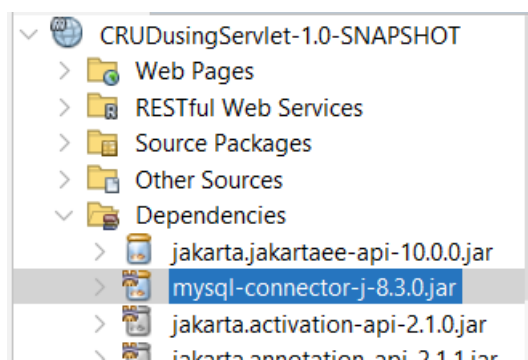
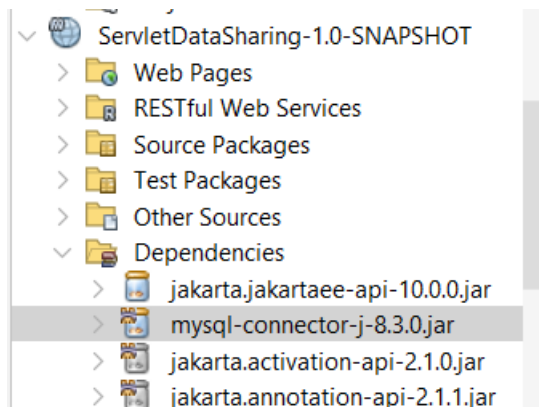
- Group ID:** mysql-connector-j-8.3.0
- Artifact ID:** mysql-connector-j
- Version:** 8.3.0
- Scope:** compile (dropdown menu)
- Type:** (empty text field)
- Classifier:** (empty text field)
- Search:** A tabbed interface with "Open Projects" and "Dependency Management" tabs. The "Search" tab is active, showing a "Query:" text field with a placeholder "(coordinate, class name, project name...)" and a "Search Results:" area below it.
- Buttons:** "Add" and "Cancel" buttons at the bottom right.



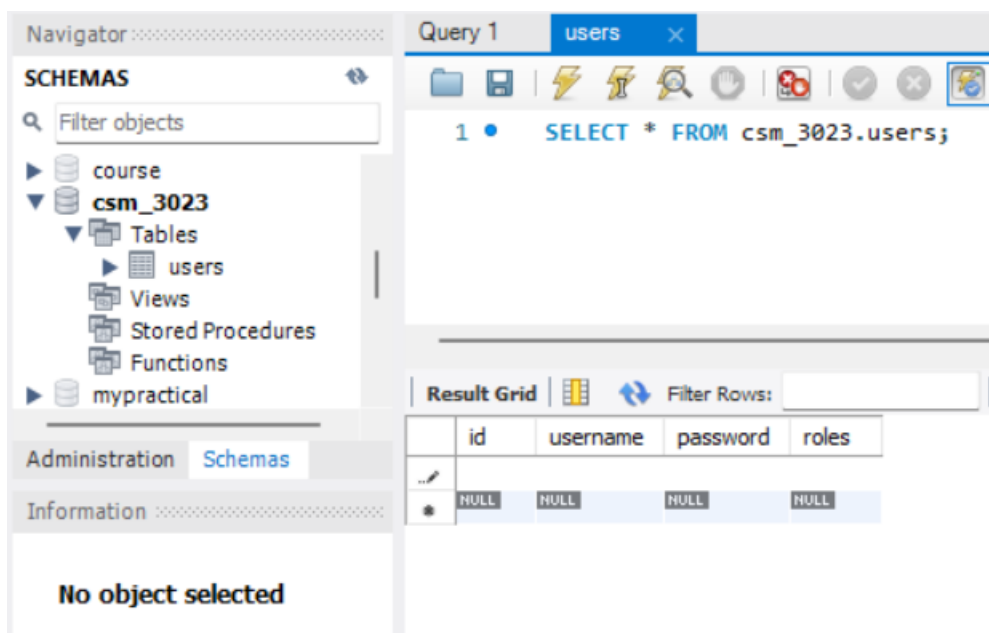
The "Install locally" dialog box is used to install a dependency from a local JAR file. It contains the following fields and options:

- Artifact to Install:** C:\Program Files\Java\library\mysql-connector-j-8.3.0.jar
- Browse...** button next to the artifact path.
- Instructions:** "Please enter path to JAR file that represents the dependency with groupId: **mysql-connector-j-8.3.0**, artifactId: **mysql-connector-j**, version: **8.3.0**. This JAR will be copied to your local repository. Please note that having non-repository based JARs as dependencies is bad for build reproducibility."
- Buttons:** "Install locally" and "Cancel" buttons at the bottom right.





### Task 3 : Setting the Environment of Web Application for Database Connection





## Task 4 : Using Servlets for Database CRUD Operations

### Index.html

```
<!DOCTYPE html>

<html>

  <head>

    <title>Start Page</title>

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

  </head>

  <body>

    <h1>Add New User</h1>

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

      <table>

        <tr><td>Username:</td><td><input type="text" name="name"/></td></tr>

        <tr><td>Password:</td><td><input type="password" name="password"/></td></tr>

        <tr><td>Role:</td><td>

          <select name="role" style="width:150px">

            <option>admin</option>

            <option>user</option>

          </select>

        </td></tr>

        <tr><td colspan="2"><input type="submit" value="Save User"/></td></tr>

      </table>

    </form>
```

```
<br/>

<a href="ViewServlet">view users</a>


</body>

</html>
```

## DeleteServlet.java

```
import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @Nur Ezreena
 */

public class DeleteServlet extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");

        String sid = request.getParameter("id");
```

```
int id = Integer.parseInt(sid);

UserDao.delete(id);

response.sendRedirect("ViewServlet");

}
```

## EditServlet.java

```
import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @Nur Ezreena
 */

public class EditServlet extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");

        PrintWriter out = response.getWriter();

        out.println("<h1>Update User</h1>");

        String sid = request.getParameter("id");
```

```
int id = Integer.parseInt(sid);

User e = UserDao.getUserById(id);

out.print("<form action='EditServlet2' method='post'>");

out.print("<table>");

out.print("<tr><td></td><td><input type='hidden' name='id' value='"

        + e.getId() + "'></td></tr>"

);

out.print("<tr><td>Name:</td><td><input type='text' name='username' value='"

        + e.getUsername() + "'></td></tr>"

);

out.print("<tr><td>Password:</td><td><input type='password' name='password' value='"

        + e.getPassword() + "'></td></tr>"

);

out.print("<tr><td>Role:</td><td>");

out.print("<select name='role' style='width:150px'>");

out.print("<option>Admin</option>");

out.print("<option>User</option>");

out.print("</select>");

out.print("</td></tr>");

out.print("<tr><td colspan='2'><input type='submit' value='Edit & Save'></td></tr>");

out.print("</table></form>");
```

```
out.close();

}
```

## EditServlet2.java

```
import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @Nur Ezreena
 */

public class EditServlet2 extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");

        PrintWriter out = response.getWriter();

        int id = Integer.parseInt(request.getParameter("id"));

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

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

```
String role = request.getParameter("role");

User e = new User();

e.setId(id);

e.setUsername(username);

e.setPassword(password);

e.setRole(role);

int status = UserDao.update(e);

if (status > 0) {

    out.print("<p>Record saved successfully!</p>");

    request.getRequestDispatcher("index.html").include(request, response);

}

else {

    out.println("Sorry! Unable to save record");

}

out.close();

}
```

## SaveServlet.java

```
import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;
```



```
import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @Nur Ezreena
 */

public class SaveServlet extends HttpServlet {

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

        response.setContentType("text/html;charset=UTF-8");

        PrintWriter out = response.getWriter();

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

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

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

        User e = new User();

        e.setUsername(name);

        e.setPassword(password);

        e.setRole(role);

        int status = UserDao.save(e);
```

```
    if (status > 0) {

        out.print("<p>Record saved successfully!</p>");

        request.getRequestDispatcher("index.html").include(request, response);

    }

    else {

        out.println("Sorry! Unable to save record.");

    }


    out.close();

}
```

## User.java

```
/**
 *
 * @Nur Ezreena
 */
public class User {

    private int id;

    private String username, password, role;

    public int getId() {

        return id;

    }

}
```

```
public void setId(int id) {
```

```
    this.id = id;
```

```
}
```

```
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;
```

```
}
```

```
public String getRole() {
```

```
    return role;
```

```
}
```

```
public void setRole(String role) {  
  
    this.role = role;  
  
}  
  
}
```

## UserDao.java

```
import java.util.*;  
  
import java.sql.*;  
  
public class UserDao {  
  
    public static Connection getConnection() {  
  
        Connection con = null;  
  
        try {  
  
            Class.forName("com.mysql.jdbc.Driver");  
  
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/csm_3023", "root",  
"admin");  
  
        }  
  
        catch (Exception e) {  
  
            System.out.println(e);  
  
        }  
  
        return con;  
  
    }  
}
```

```
public static int save(User e) {

    int status = 0;

    try {

        Connection con = UserDao.getConnection();

        PreparedStatement ps = con.prepareStatement(

            "INSERT INTO users(username, password, roles) VALUES (?, ?, ?)"

        );

        ps.setString(1, e.getUsername());

        ps.setString(2, e.getPassword());

        ps.setString(3, e.getRole());

        status = ps.executeUpdate();

        con.close();

    } catch (Exception ex) {

        ex.printStackTrace();

    }

    return status;

}

public static int update(User e) {

    int status = 0;

    try {
```

```
        Connection con = UserDao.getConnection();

        PreparedStatement ps = con.prepareStatement(

            "UPDATE users SET username = ?, password = ?, roles = ? WHERE id = ?"

        );

        ps.setString(1, e.getUsername());

        ps.setString(2, e.getPassword());

        ps.setString(3, e.getRole());

        ps.setInt(4, e.getId());


        status = ps.executeUpdate();


        con.close();
    } catch (Exception ex) {

        ex.printStackTrace();

    }


    return status;

}


public static int delete(int id) {

    int status = 0;


    try {

        Connection con = UserDao.getConnection();
```

```
        PreparedStatement ps = con.prepareStatement(

            "DELETE FROM users WHERE id = ?"

        );

        ps.setInt(1, id);

        status = ps.executeUpdate();

        con.close();

    } catch (Exception ex) {

        ex.printStackTrace();

    }

    return status;

}
```

```
public static User getUserById(int id) {

    User e = new User();

    try {

        Connection con = UserDao.getConnection();

        PreparedStatement ps = con.prepareStatement(

            "SELECT * FROM users WHERE id = ?"

        );

        ps.setInt(1, id);
```

```
        ResultSet rs = ps.executeQuery();

        if (rs.next()) {

            e.setId(rs.getInt(1));

            e.setUsername(rs.getString(2));

            e.setPassword(rs.getString(3));

            e.setRole(rs.getString(4));

        }

        con.close();

    } catch (Exception ex) {

        ex.printStackTrace();

    }

    return e;

}

public static List<User> getAllUsers() {

    List<User> list = new ArrayList<User>();

    try {

        Connection con = UserDao.getConnection();

        PreparedStatement ps = con.prepareStatement(

            "SELECT * FROM users"

        );

    }
```



```

        ResultSet rs = ps.executeQuery();

        while (rs.next()) {

            User e = new User();

            e.setId(rs.getInt(1));

            e.setUsername(rs.getString(2));

            e.setPassword(rs.getString(3));

            e.setRole(rs.getString(4));

            list.add(e);

        }

        con.close();

    } catch (Exception ex) {

        ex.printStackTrace();

    }

    return list;

}

}

```

## ViewServlet.java

```

import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

```

```

import jakarta.servlet.http.HttpServletResponse;

import java.util.List;

/**
 *
 * @Nur Ezreena
 */

public class ViewServlet extends HttpServlet {

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

        response.setContentType("text/html;charset=UTF-8");

        PrintWriter out = response.getWriter();

        out.println("<a href='index.html'>Add New User</a>");

        out.println("<h1>User List</h1>");

        List<User> list = UserDao.getAllUsers();

        out.print("<table border='1' width='100%'>");

        out.print("<tr><th>Id</th><th>Name</th><th>Password</th><th>Role</th><th>Edit</th><th>D
        elete</th>");

        for (User e : list) {

            out.print("<tr><td>" + e.getId() + "</td><td>" + e.getUsername() + "</td><td>"

                + e.getPassword() + "</td><td>" + e.getRole() + "</td><td><a href='EditServlet?id="

```

```

        + e.getId() + ">Edit</a></td><td><a href='DeleteServlet?id="
        + e.getId() + ">Delete</a></td></tr>"

    );
}

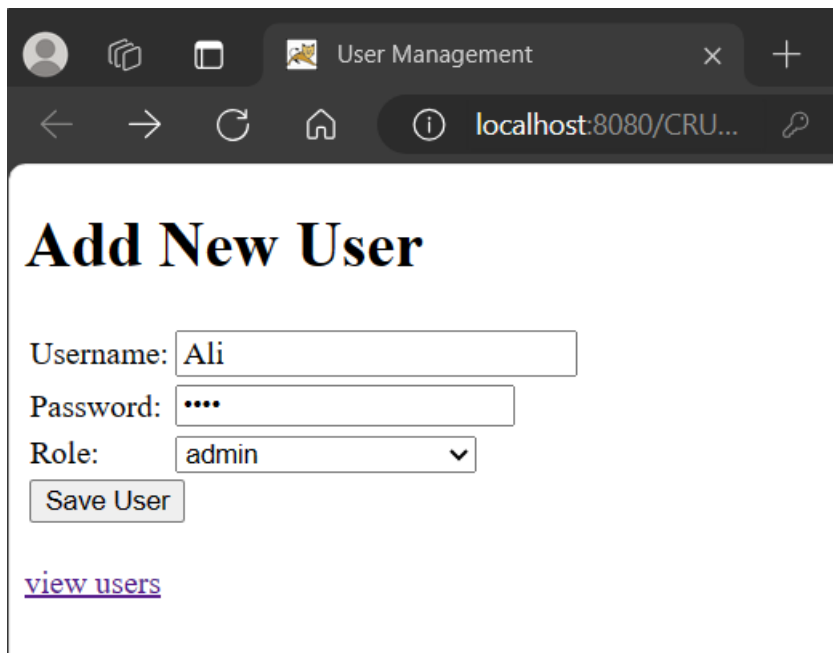
out.print("</table>");

out.close();

}

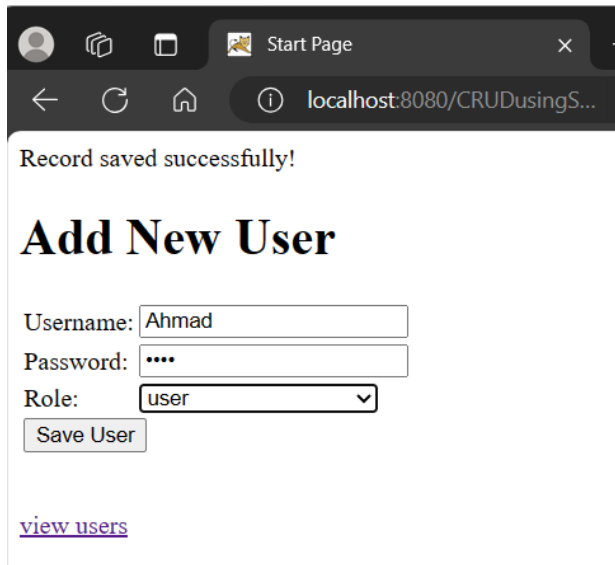
```

Output:



The screenshot shows a web browser window with the title 'User Management'. The address bar displays 'localhost:8080/CRU...'. The main content area features a heading 'Add New User' in a large, bold, black serif font. Below the heading is a form with three input fields: 'Username' containing the text 'Ali', 'Password' with masked characters '....', and 'Role' with a dropdown menu showing 'admin'. A 'Save User' button is positioned below the 'Role' field. At the bottom of the form, there is a purple underlined link labeled 'view users'.

Add:



Record saved successfully!

## Add New User

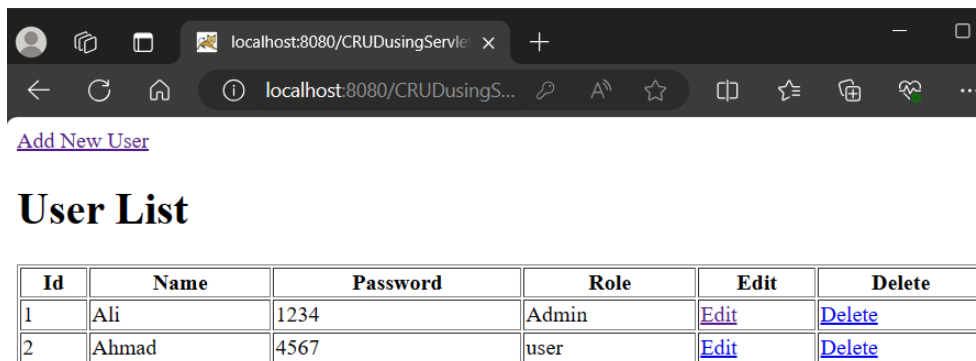
Username:

Password:

Role:

[view users](#)

View user:

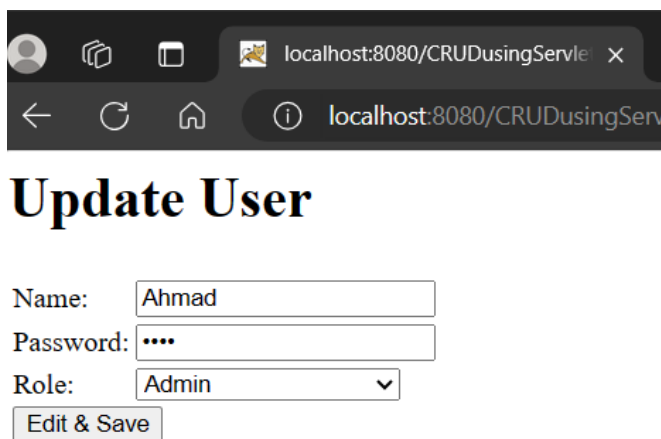


[Add New User](#)

## User List

Id	Name	Password	Role	Edit	Delete
1	Ali	1234	Admin	<a href="#">Edit</a>	<a href="#">Delete</a>
2	Ahmad	4567	user	<a href="#">Edit</a>	<a href="#">Delete</a>

Edit:



## Update User

Name:

Password:

Role:

localhost:8080/CRUDusingServlet/...

←

↺

🏠

🔍

🌟

📄

🔖

🔗

🔧

⋮

[Add New User](#)

## User List

<b>Id</b>	<b>Name</b>	<b>Password</b>	<b>Role</b>	<b>Edit</b>	<b>Delete</b>
1	Ali	1234	Admin	<a href="#">Edit</a>	<a href="#">Delete</a>
2	Ahmad	4567	Admin	<a href="#">Edit</a>	<a href="#">Delete</a>

Delete:

localhost:8080/CRUDusingServlet/...

←

↺

🏠

🔍

🌟

📄

🔖

🔗

🔧

⋮

[Add New User](#)

## User List

<b>Id</b>	<b>Name</b>	<b>Password</b>	<b>Role</b>	<b>Edit</b>	<b>Delete</b>
2	Ahmad	4567	Admin	<a href="#">Edit</a>	<a href="#">Delete</a>