

UNIVERSITI MALAYSIA TERENGGANU

CSM3023 WEB BASED APPLICATION DEVELOPMENT (K1)

BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONORS

LAB REPORT

LAB 2 - Servlet: Data Sharing and Database Management

SEMESTER II 2023/2024

Prepared for:

Dr MOHAMAD NOR BIN HASAN

Prepared by:

IZZUL ILYAS BIN ROSLAN (S65343)

Task 1: Data Sharing in Servlet

Objective: To use servlet for request forwarding and data sharing Problem Description: Write a login form and a servlet to authenticate a user

Code:

login .html

```
Projects × Files Services _ Start Page × 🕝 login.html × 🗷 LoginServlet.java × 🗷 AccountServlet.java ×
  ServiceDataSharing-1.0-SNAPSHC

ServiceDataSharing-1.0-SNAPSHC

Web Pages
                                      META-INF
WEB-INF
       index.html
                                                       <title>Login Page</title>
                                                       <meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
    RESTful Web Services
Source Packages
       <default package>

    AccountServlet.java
    LoginServlet.java

                                                            body {
                                                                 background-color: black;
       com.mycompany.servletda
                                                                 text-align: left;
color: white;
     Test Packages
                                                                 font-family: Arial, Helvetica, sans-serif;
     Other Sources
                                                        </style>
                                                  </head>
Navigator ×
   Thi. CSS
                                                        <h1>Welcome to CSM3023</h1>
  HTML
                                                       Please insert your username and password
<form name="loqin" id="loqin" action="loqinServlet" method="POST" autoo
Username:<input name="txtUsername" type="text"> <br>
Password:<input name="txtBassword" type="text"> <br>
Password:<input name="txtBassword" type="text"> <br>

     ♠ html
       € head
       body

h1
                                      22

    p
    form id=login

                                      25
26
27
                                                             <input name="txtReset" value="Reset" type="reset"><br/>br
             input br
                                                       </form
                                                       <br>
                                      28
              €≫ br
                                                   </body>
                                              </html>
                                      31
              input
☐ 万 Output
                                                                                                                                                                               1
                                                                                                                                                                                         24:62
                                                                                                                                                                                                    INS Unix (LF)
```

Login.servlet

```
Projects × Files Services _
                                Start Page × 🕝 login.html × 🚳 LoginServlet.java × 🚳 AccountServlet.java ×
  ServletDataSharing-1.0-SNAPSHC Source History 🖟 😽 📲 - 🔍 👼 🖶 📑 🖟 😓 🗟 2 1 0 0 1 4 1
    Web Pages
                                       import java.util.HashMap;
import jakarta.servlet.ServletException;
                                       import jakarta.servlet.http.HttpServlet;
                                       import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
    RESTful Web Services
Source Packages
                                       import jakarta.servlet.RequestDispatcher;
                                       import jakarta.servlet.ServletContext;
                                       public class LoginServlet extends HttpServlet {
                                           HashMap<String, String> users = new HashMap<>();
    Test Packages
    Other Sources
                                           public void init() throws ServletException {
LoginServlet.java - Navigator × ___
                                               super.init();
                                                users.put("Ali", "1234");
users.put("Ahmad", "4567");
users.put("Muthu", "8910");
            ✓ <empty> ✓ 📅
  ▲ LoginServlet :: HttpServlet

    ♦ LoginServlet()
    ♦ doPost(HttpServletRequest requ
     init() ↑ GenericServlet
     processRequest(HttpServletRequ
                                           protected void doPost(HttpServletRequest request, HttpServletResponse response)
    users: HashMap<String, String>
                                                     throws ServletException, IOException {
                                                processRequest(request, response);
                                            protected void processRequest(HttpServletRequest request, HttpServletResponse response)
                                                response.setContentType("text/html;charset=UTF-8");
□ □ Output
```

```
Source History | 🔀 📴 🔻 🔻 🔻 🖓 🐉 🚭 | 🚱 😂 🔮 | • □ | 😃 🛓
    📸 Web Pages
                                             protected void processRequest(HttpServletRequest request, HttpServletResponse response)
       META-INF
WEB-INF
                                                       throws ServletException, IOException {
       index.html
                                                  response.setContentType("text/html;charset=UTF-8");
                                                  String username = request.getParameter("txtUsername");
     RESTful Web Services
                                       String password = request.getParameter("txtPassword");
    Source Packages

default package>
AccountServlet.java
LoginServlet.java
                                                 if (!username.equals("") && !password.equals("")
                                                      1 Set users.containsKey(username) & users.get(username).equals(password)) {
    request.setAttribute("useria", username);
    ServletContext sc = getServletContext();
    RequestDispatcher rd = sc.getRequestDispatcher("/AccountServlet");
    com.mycompany.servletd:
com.mycompany.servletd:
Test Packages
                                                      rd.forward(request, response);
    Other Sources
                                                 } else {
processRequest - Navigator ×
                                                       RequestDispatcher rd = request.getRequestDispatcher("/login.html");
            ✓ <empty> ✓ 😈
Members
                                                      rd.forward(request, response);
init() † GenericServlet
processRequest(HttpServletRequest
users : HashMap<String, String>
🗇 👨 Output
                                                                                                                                                            33:1
                                                                                                                                                                              INS Windows (CRLF)
```

Account.servlet

```
Project x Files Services

Sant Page x 3 Jogn.html x 3 LoginServletjava x

ServictLabsharing-10-NSAPSH

ServictLabsharing-10-NSAPSH
```

```
Projects X Files Services ___ Start Page X @ login.html X @ LoginServlet.java X @ AccountServlet.java X
  © ServictDatsharing-1.0-SNAPSHC

© ServictDatsharing-1.0-SNAPSHC

© Web Pages

© Devictor void processRequest (HttpServletRequest request, H
                                                protected void processRequest(HttpServletRequest request, HttpServletResponse response)
                                      9
34 □
       META-INF
WEB-INF
                                                       throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
                                      35
36
37
38
39
        index.html
                                                       String userid_login = (String) request.getAttribute("userid");
     RESTful Web Services
Source Packages
                                                      try (PrintWriter out = response.getWriter()) {
      40
41
42

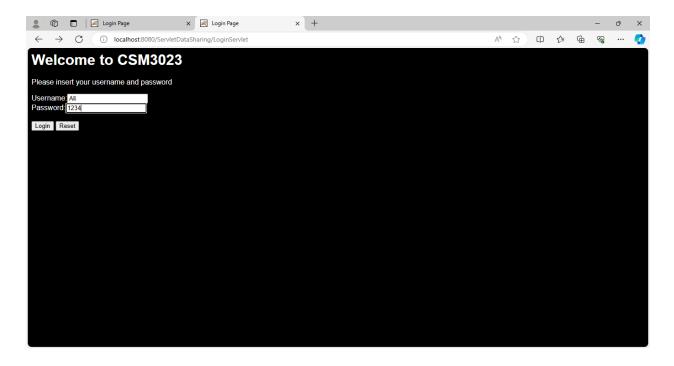
    AccountServlet.java
    LoginServlet.java

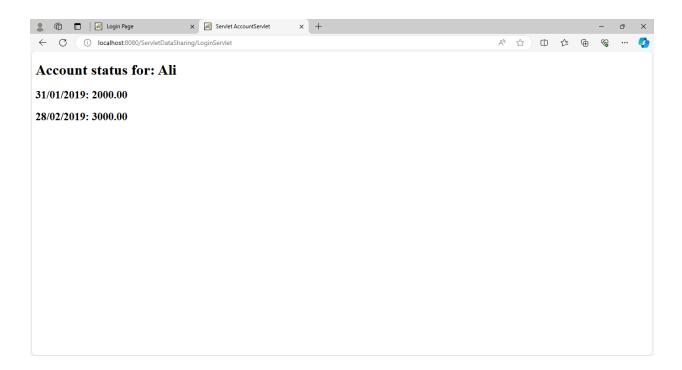
                                                            out.println("<!DOCTYPE html>");
out.println("<html>");
out.println("<head>");
                                                            out.println("<title>Servlet AccountServlet</title>");
out.println("</head>");
                                                            out.println("<body>");
                                                            if (account.get(userid_login) == null) {
                                                                 out.println("<hl>Sorry, no information found!</hl>");
▲ AccountServlet :: HttpServlet
                                                                out.println("<hl>Account status for: " + userid login + "</hl>");

    ♦ AccountServlet()
    ♦ doGet(HttpServletRequest request,

                                                                 for (String tempAcc : account.get(userid_login)) {
   out.println("<h2>" + tempAcc + "</h2>");
  oinit() † GenericServletRequest request,
processRequest(HttpServletRequest
                                      55
56
57
58
59
60
61
62
   account : HashMap < String, String[]
                                                            out.println("</body>");
                                                            out.println("</html>");
 ☐ 👨 Output
                                                                                                                                                                              1
                                                                                                                                                                                       33:14
                                                                                                                                                                                                  INS Windows (CRLF)
```

Output:





Reflection

1. What have you you learnt from this exercise?

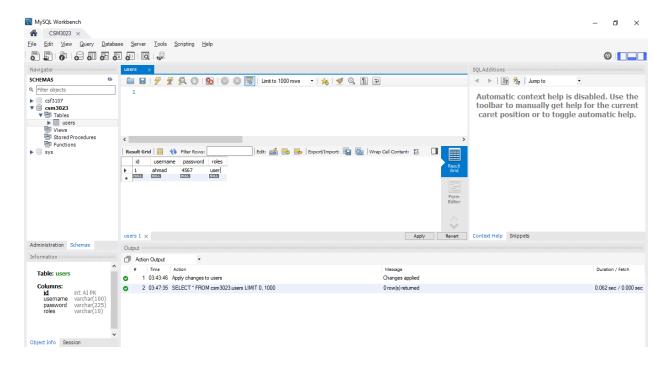
On how to use servlet to get the data

2. What are the common methods used in Java Servlet?

doPost(), doGet()

Task 2: Creating A Table in MySQL Database

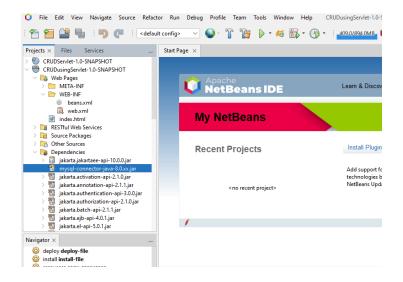
Objective: To create a MySQL table to store user credentials Problem Description: Prepare a user table to be used in Web Application



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

Objective: To set up a proper environment for integrating web application to the database

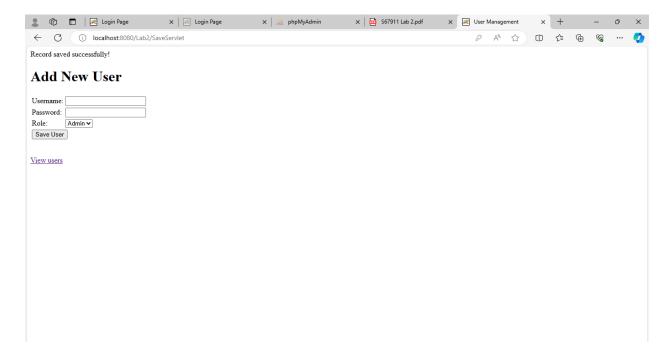
Problem Description: Import MySQL JDBC Library to an existing project

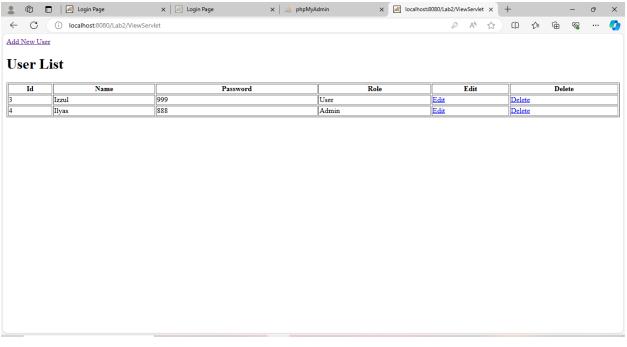


Task 4: Using Servlets for Database CRUD Operations

Objective: To program multiple servlets for manipulating the database Problem Description: Program five different servlets to handle database operations such as insert, update and delete.

- i. SaveServlet.java: to save data into the database
- ii. ViewServlet. java: to view data retrieved from database
- iii. EditServlet.java & EditServlet2.java: to edit existing data
- iv. DeleteServlet.java: to delete existing data Apart from the servlets, we are going to develop two custom Java class know as JavaBeans and Data Access Object (DAO)







Reflections:

1. What is the name of the Java Library that you need to import before coding the web application with database operations?

JDBC(Java Database Connectivity)

- 2. Which folder keeps the web.xml file? Copy the contents of the file and explain in brief the tags included such as
- <servlet-name><servlet-class><servlet-mapping>. etc.
- <servlet-name> specifies a unique name for the servlet configuration.
- <servlet-class> specifies the fully qualified class name of the servlet implementation.
- <servlet-mapping> maps a servlet to a URL pattern. It defines the URLs that invoke the servlet.
- <url>pattern>specifies the URL pattern to which the servlet is mapped.
- 3. Define the usage of Data Access Object (DAO) servlet. How it ease the business process in your servlet-based web application?

A Data Access Object (DAO) servlet is a design pattern used to separate the data access logic from the business logic of a servlet-based web application. The DAO pattern involves creating a separate class or set of classes responsible for interacting with the database, querying data, and performing database operations. By using a DAO servlet, the business logic in servlets can focus on handling user requests, processing data, and generating responses, while the data access logic is encapsulated within the DAO classes, improving the overall organization and clarity of the application architecture.