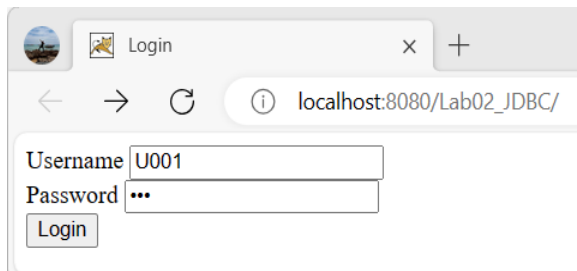# Building an Java Web Application
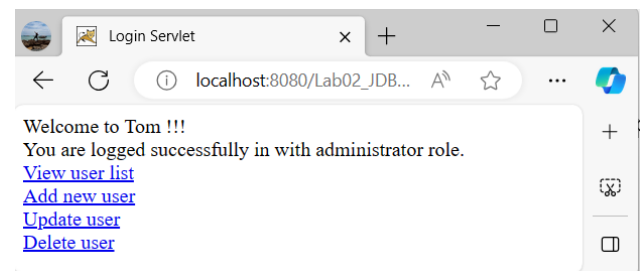
Using HTML, Servlet & JDBC

Requirements

Create a java web application (using HTML, Servlets and JDBC) to perform functions as login and manage users (CRUD) for admin role
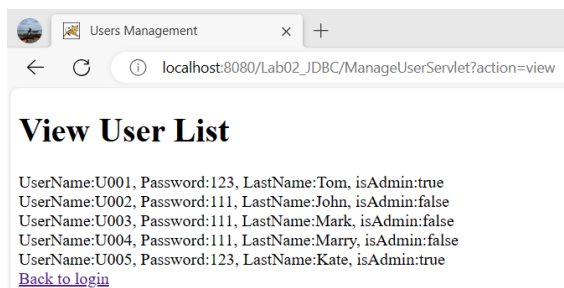

Lab Objectives

In this lab, you will be:

- Create Java web project using the NetBean IDE
- Create the web pages by HTML
- Apply passing data from HTML to Servlet by GET and POST method

- Performs CRUD functions by SqlJDBC driver and SQL Server
- Run the project and test the functions of the application

Step 01. Create a Java Web application named **IntroductionJDBC** is structured as follows:
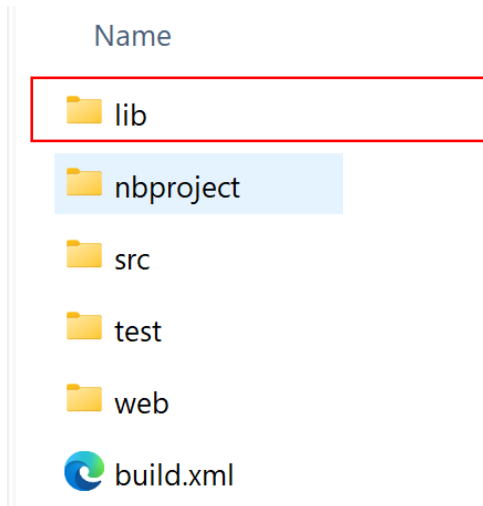
Create folder named **lib** in the project (use File Explorer on Windows) and copy sqljdbc4.jar that match with SQL Server version on the machine to **lib** folder
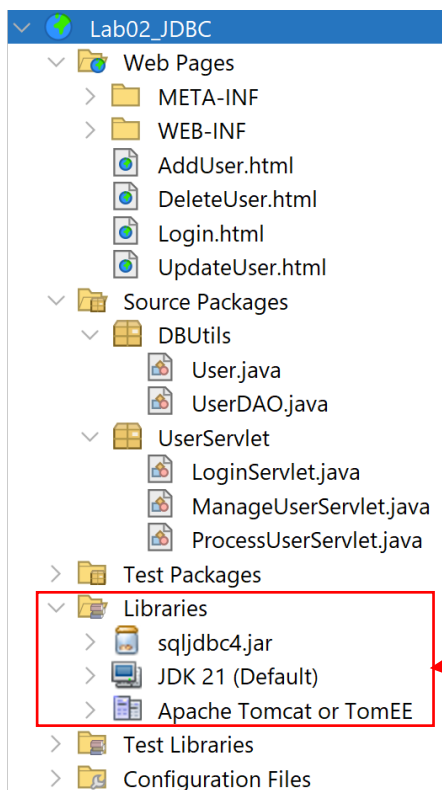
Right-click on the project, select **Properties | Libraries | Add JAR/Folder** then select **sqljdbc4** driver in the **lib** folder that matches the Sql Server version on the machine.

-Open SQL Server Management Studio and create a database named **SampleDB** has a table named **Registration** as follows:

Column Name	Data Type	Allow Nulls
🔑 UserName	varchar(30)	☐
Password	varchar(30)	☐
LastName	nvarchar(50)	☐
isAdmin	bit	☐

UserName	Password	LastName	isAdmin
U001	123	Tom	True
U002	456	David	False
U003	789	John	False
U004	012	Mark	True
U005	134	Kate	False

Step 02. Write codes for the classes in the **DBUtils** package as the follows:

- **User.java**

```java
package DBUtils;

/**
 * @author SwordLake
 */
public class User {
    private String userName;
    private String password;
    private String lastname;
    private boolean isAdmin;
    public User() {
        this.userName = null;
        this.password = null;
        this.lastname = null;
        this.isAdmin = false;
    }
    public User(String userName, String password, String lastname, boolean isAdmin)
        this.userName = userName;
        this.password = password;
        this.lastname = lastname;
        this.isAdmin = isAdmin;
    }
    public String getUserName() {...3 lines }
    public void setUserName(String userName) {...3 lines }
    public String getPassword() {...3 lines }
    public void setPassword(String password) {...3 lines }
    public String getLastname() {...3 lines }
    public void setLastname(String lastname) {...3 lines }
    public boolean isIsAdmin() {...3 lines }
    public void setIsAdmin(boolean isAdmin) {...3 lines }
    @Override
    public String toString() {
        return String.format("UserName:%s, Password:%s, LastName:%s, isAdmin:%b",
                userName,password,lastname,isAdmin);
    }
}//end class
```

- **UserDAO.java**

```java
7    import java.sql.SQLException;
8    import java.util.ArrayList;
9    import java.util.List;
10
11   /**
12    * @author SwordLake
13    */
14   public class UserDAO {
15       // <editor-fold defaultstate="collapsed" desc="getConnection Method">
16       public Connection getConnection() throws Exception {
17           try {
18               Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
19               String connectionString =
20               "jdbc:sqlserver://localhost:1433;database=SampleDB;instanceName=SQL2019";
21               //SQL Server Authentication
22               Connection cnn = DriverManager.getConnection(connectionString, "sa", "123");
23               return cnn;
24           } catch (ClassNotFoundException | SQLException ex) {
25               throw ex;
26           }
27       }
28       // </editor-fold>

30       // <editor-fold defaultstate="collapsed" desc="login Method">
31       public User login(String userName, String password) throws Exception {
32           User user = null;
33           Connection cnn = null;
34           PreparedStatement preStm = null;
35           ResultSet rs = null;
36           String lastName;
37           boolean isAdmin;
38           try {
39               cnn = getConnection();
40               String sql =
41               "select LastName,isAdmin from Registration where [UserName]=? and [Password]=?";
42               preStm = cnn.prepareStatement(sql);
43               preStm.setString(1, userName);
44               preStm.setString(2, password);
45               rs = preStm.executeQuery();
46               while (rs.next()) {
47                   lastName = rs.getString(1);
48                   isAdmin = rs.getBoolean(2);
49                   user = new User(userName, password, lastName, isAdmin);
50               }//end while
51           } catch (Exception ex) {
52               throw ex;
53           } finally {
54               if (rs != null) {
55                   rs.close();
56               }
57               if (preStm != null) {
58                   preStm.close();
59               }
```

```java
 60              if (cnn != null) {
 61                  cnn.close();
 62              }
 63          }
 64          return user;
 65      }//end login
 66      // </editor-fold>
 67      //------------------------------------------------------
 68      // <editor-fold defaultstate="collapsed" desc="getUserList Method">
 69      public List<User> getUserList() throws Exception {
 70          String userName, password, lastName;
 71          boolean isAdmin;
 72          Connection cnn = null;
 73          PreparedStatement preStm = null;
 74          ResultSet rs = null;
 75          List<User> userList = new ArrayList();
 76          try {
 77              cnn = getConnection();
 78              String sql = "select UserName,Password,LastName,isAdmin from Registration";
 79              preStm = cnn.prepareStatement(sql);
 80              rs = preStm.executeQuery();
 81              while (rs.next()) {
 82                  userName = rs.getString(1);
 83                  password = rs.getString(2);
 84                  lastName = rs.getString(3);
 85                  isAdmin = rs.getBoolean(4);
 86                  User user = new User(userName, password, lastName, isAdmin);
 87                  userList.add(user);
 88              }//end while
 89          } catch (Exception ex) {
 90              throw ex;
 91          } finally {
 92              if (rs != null) {
 93                  rs.close();
 94              }
 95              if (preStm != null) {
 96                  preStm.close();
 97              }
 98              if (cnn != null) {
 99                  cnn.close();
100              }
101          }
102          if (userList.isEmpty()) {
103              return null;
104          }
105          return userList;
106      }//end getUserList
107      // </editor-fold>
108      //------------------------------------------------------
109      // <editor-fold defaultstate="collapsed" desc="addUser Method">
110      public boolean addUser(User user) throws Exception {
111          PreparedStatement preStm = null;
112          Connection cnn = null;
```

```java
113            try {
114                cnn = getConnection();
115                String sql = "Insert Registration values(?,?,?,?)";
116                preStm = cnn.prepareStatement(sql);
117                preStm.setString(1, user.getUserName());
118                preStm.setString(2, user.getPassword());
119                preStm.setString(3, user.getLastname());
120                preStm.setBoolean(4, user.isIsAdmin());
121                return preStm.executeUpdate() > 0;
122            } catch (Exception ex) {
123                throw ex;
124            } finally {
125                if (preStm != null) {
126                    preStm.close();
127                }
128                if (cnn != null) {
129                    cnn.close();
130                }
131            }
132        }//end addUser
133        // </editor-fold>
134        //-----------------------------------------------------
135        // <editor-fold defaultstate="collapsed" desc="updateUser Method">
136        public boolean updateUser(User user) throws Exception {
137            PreparedStatement preStm = null;
138            Connection cnn = null;
139            try {
140                cnn = getConnection();
141                String sql =
142                "Update Registration Set Password=?,LastName=?,isAdmin=? Where UserName=?";
143                preStm = cnn.prepareStatement(sql);
144                preStm.setString(1, user.getPassword());
145                preStm.setString(2, user.getLastname());
146                preStm.setBoolean(3, user.isIsAdmin());
147                preStm.setString(4, user.getUserName());
148                return preStm.executeUpdate() > 0;
149            } catch (Exception ex) {
150                throw ex;
151            } finally {
152                if (preStm != null) {
153                    preStm.close();
154                }
155                if (cnn != null) {
156                    cnn.close();
157                }
158            }
159        }//end updateUser
160        // </editor-fold>
161        //-----------------------------------------------------
162        // <editor-fold defaultstate="collapsed" desc="deleteUser Method">
163        public boolean deleteUser(String userName) throws Exception {
164            PreparedStatement preStm = null;
165            Connection cnn = null;
166            try {
167                cnn = getConnection();
168                String sql = "delete Registration Where UserName=?";
```

```
169                   preStm = cnn.prepareStatement(sql);
170                   preStm.setString(1, userName);
171                   return preStm.executeUpdate() > 0;
172             } catch (Exception ex) {
173                   throw ex;
174             } finally {
175                   if (preStm != null) {
176                         preStm.close();
177                   }
178                   if (cnn != null) {
179                         cnn.close();
180                   }
181             }
182       }//end deleteUser
183       // </editor-fold>
184   }//end class
```
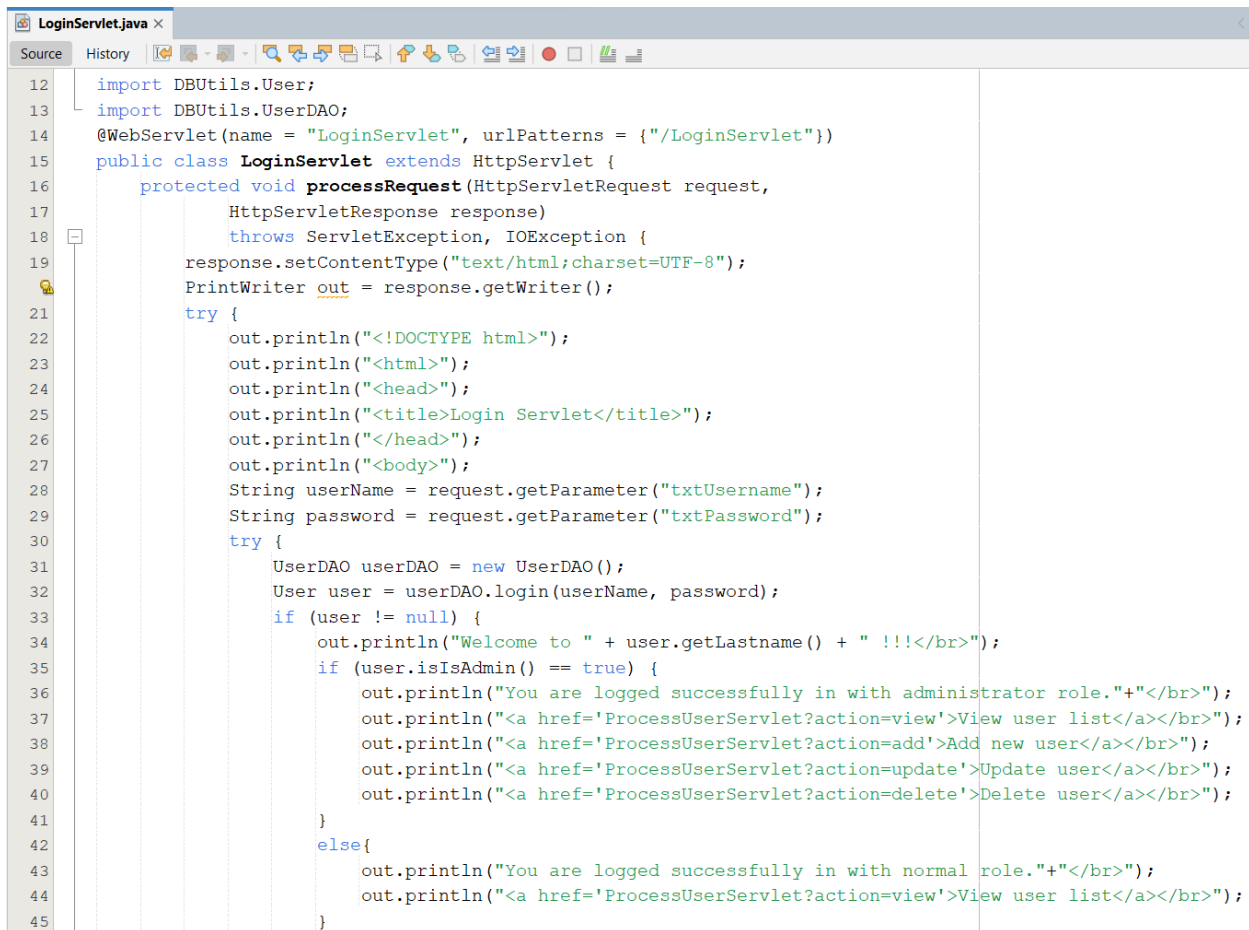
Step 03. Write codes for the classes in the **UserServlet** package as the follows:

- **LoginServlet.java**

```
LoginServlet.java ×
Source  History

12    import DBUtils.User;
13    import DBUtils.UserDAO;
14    @WebServlet(name = "LoginServlet", urlPatterns = {"/LoginServlet"})
15    public class LoginServlet extends HttpServlet {
16        protected void processRequest(HttpServletRequest request,
17                HttpServletResponse response)
18                throws ServletException, IOException {
19            response.setContentType("text/html;charset=UTF-8");
20            PrintWriter out = response.getWriter();
21            try {
22            out.println("<!DOCTYPE html>");
23            out.println("<html>");
24            out.println("<head>");
25            out.println("<title>Login Servlet</title>");
26            out.println("</head>");
27            out.println("<body>");
28            String userName = request.getParameter("txtUsername");
29            String password = request.getParameter("txtPassword");
30            try {
31                UserDAO userDAO = new UserDAO();
32                User user = userDAO.login(userName, password);
33                if (user != null) {
34                    out.println("Welcome to " + user.getLastname() + " !!!</br>");
35                    if (user.isIsAdmin() == true) {
36                        out.println("You are logged successfully in with administrator role."+"</br>");
37                        out.println("<a href='ProcessUserServlet?action=view'>View user list</a></br>");
38                        out.println("<a href='ProcessUserServlet?action=add'>Add new user</a></br>");
39                        out.println("<a href='ProcessUserServlet?action=update'>Update user</a></br>");
40                        out.println("<a href='ProcessUserServlet?action=delete'>Delete user</a></br>");
41                    }
42                    else{
43                        out.println("You are logged successfully in with normal role."+"</br>");
44                        out.println("<a href='ProcessUserServlet?action=view'>View user list</a></br>");
45                    }
```

```
46                        } else {
47                            out.println("Loggin has failed. </br>");
48                            out.println("<a href='Login.html'>Back to login</a>");
49                        }
50                    } catch (Exception ex) {
51                        out.println("Something went wrong. Error :" + ex.getMessage());
52                    }
53                    out.println("</body>");
54                    out.println("</html>");
55                }
56                finally{
57                    out.close();
58                }
59            }
60    +      HttpServlet methods. Click on the + sign on the left to edit the code.
98        }
```
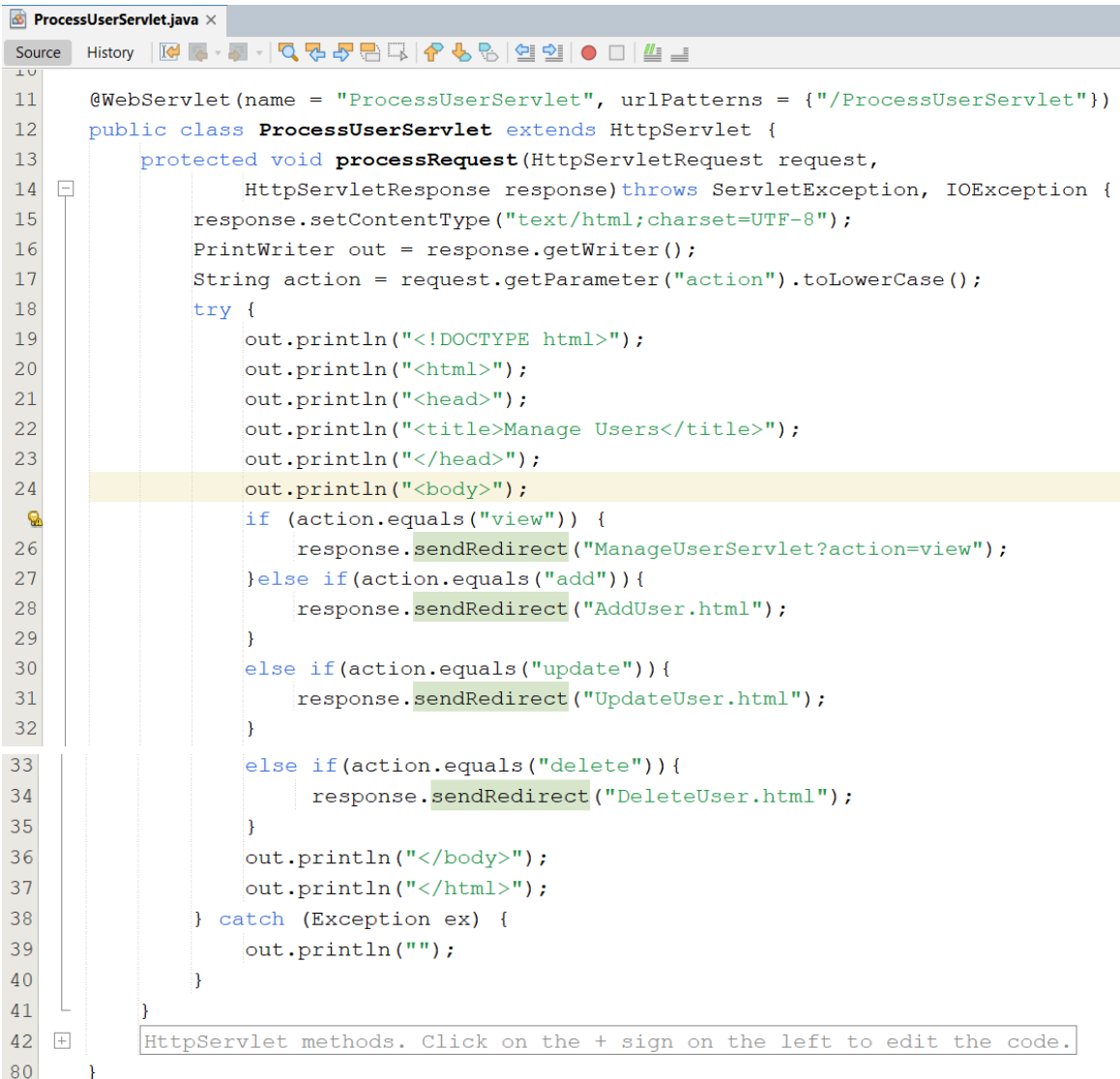
- **ProcessUserServlet.java**

```
11    @WebServlet(name = "ProcessUserServlet", urlPatterns = {"/ProcessUserServlet"})
12    public class ProcessUserServlet extends HttpServlet {
13        protected void processRequest(HttpServletRequest request,
14                HttpServletResponse response) throws ServletException, IOException {
15            response.setContentType("text/html;charset=UTF-8");
16            PrintWriter out = response.getWriter();
17            String action = request.getParameter("action").toLowerCase();
18            try {
19                out.println("<!DOCTYPE html>");
20                out.println("<html>");
21                out.println("<head>");
22                out.println("<title>Manage Users</title>");
23                out.println("</head>");
24                out.println("<body>");
              if (action.equals("view")) {
26                    response.sendRedirect("ManageUserServlet?action=view");
27                }else if(action.equals("add")){
28                    response.sendRedirect("AddUser.html");
29                }
30                else if(action.equals("update")){
31                    response.sendRedirect("UpdateUser.html");
32                }
33                else if(action.equals("delete")){
34                    response.sendRedirect("DeleteUser.html");
35                }
36                out.println("</body>");
37                out.println("</html>");
38            } catch (Exception ex) {
39                out.println("");
40            }
41        }
42    +      HttpServlet methods. Click on the + sign on the left to edit the code.
80    }
```

- ## ManageUserServlet.java

```java
import DBUtils.User;
import DBUtils.UserDAO;
import java.util.List;
@WebServlet(name = "ManageUserServlet", urlPatterns = {"/ManageUserServlet"})
public class ManageUserServlet extends HttpServlet {
    protected void processRequest(HttpServletRequest request,
            HttpServletResponse response)
            throws ServletException, IOException {
        String userName, password, lastName;
        boolean isAdmin = true;
        String action;
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        action = request.getParameter("action").toLowerCase();
        try {
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Users Management</title>");
            out.println("</head>");
            out.println("<body>");
            userName = request.getParameter("txtUserName");
            UserDAO userDAO = new UserDAO();
            if(action.equals("view")) {
                out.println("<h1>View User List" + "</h1>");
                List<User> userList = userDAO.getUserList();
                for (User user : userList) {
                    out.println(user + "</br>");
                }
            }
            else if (action.equals("delete")) {
                out.println("<h1>Users Management - Delete user" + "</h1>");
                if (userDAO.deleteUser(userName) == true) {
                    out.println("User has been deleted successfully</br>");
                } else {
                    out.println("<h3>Something went wrong!</h3></br>");
                }
            } else {
                password = request.getParameter("txtPassword");
                lastName = request.getParameter("txtLastName");
                String admin = request.getParameter("chkIsAdmin");
                if (admin == null) {
                    isAdmin = false;
                }
                User user = new User(userName, password, lastName, isAdmin);
                if (action.equals("add")) {
                    out.println("<h1>Users Management - Add new user" + "</h1>");
                    if (userDAO.addUser(user) == true) {
                        out.println("User has been added successfully</br>");
```

```
60                          } else {
61                              out.println("<h3>Something went wrong!</h3></br>");
62                          }
63                      } else if (action.equals("update")) {
64                          out.println("<h1>Users Management - Update user" + "</h1>");
65                          if (userDAO.updateUser(user) == true) {
66                              out.println("User has been updated successfully</br>");
67                          } else {
68                              out.println("<h3>Something went wrong!</h3></br>");
69                          }
70                      }
71                  }
72                  out.println("<a href='Login.html'>Back to login</a>");
73                  out.println("</body>");
74                  out.println("</html>");
75              } catch (Exception ex) {
76                  out.println("Error:" + ex.getMessage());
77              }
78          }
79      +   HttpServlet methods. Click on the + sign on the left to edit the code.
117     }//end class
```
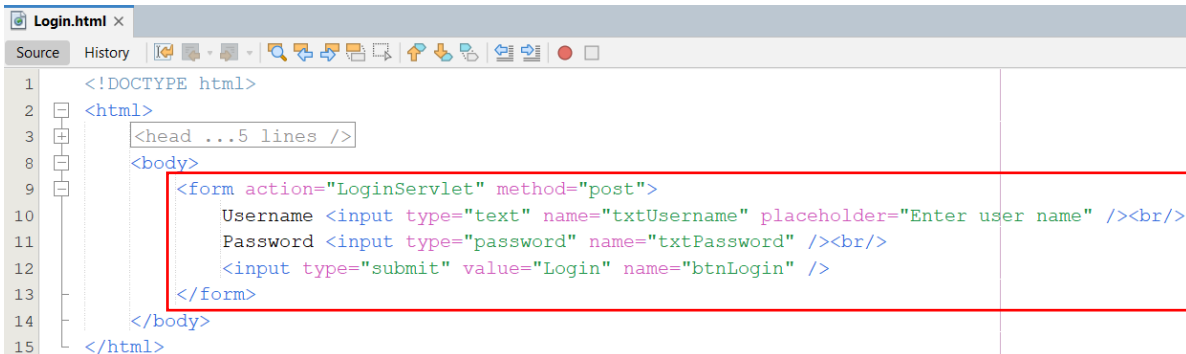
Step 04. Write HTML codes for the web pages as the follows:

- **Login.html page**

```
Login.html ×

Source  History

1    <!DOCTYPE html>
2    <html>
3        <head ...5 lines />
8        <body>
9            <form action="LoginServlet" method="post">
10               Username <input type="text" name="txtUsername" placeholder="Enter user name" /><br/>
11               Password <input type="password" name="txtPassword" /><br/>
12               <input type="submit" value="Login" name="btnLogin" />
13           </form>
14       </body>
15   </html>
```

- **AddUser.html page**

```
AddUser.html ×

Source  History

1    <!DOCTYPE html>
2    <html>
3        <head ...5 lines />
8        <body>
9            <form action="ManageUserServlet?action=add" method="post">
10               <h1>Add new user</h1>
11               UserName <input type="text" placeholder="Enter user name" name="txtUserName" /><br/>
12               Password <input type="password" placeholder="Enter password" name="txtPassword"/><br/>
13               LastName <input type="text" placeholder="Enter last name" name="txtLastName"/><br/>
14               <input type="checkbox" name="chkIsAdmin" />isAdmin<br/>
15               <input type="submit" value="Add" name="btnAdd" />
16           </form>
17       </body>
18   </html>
```
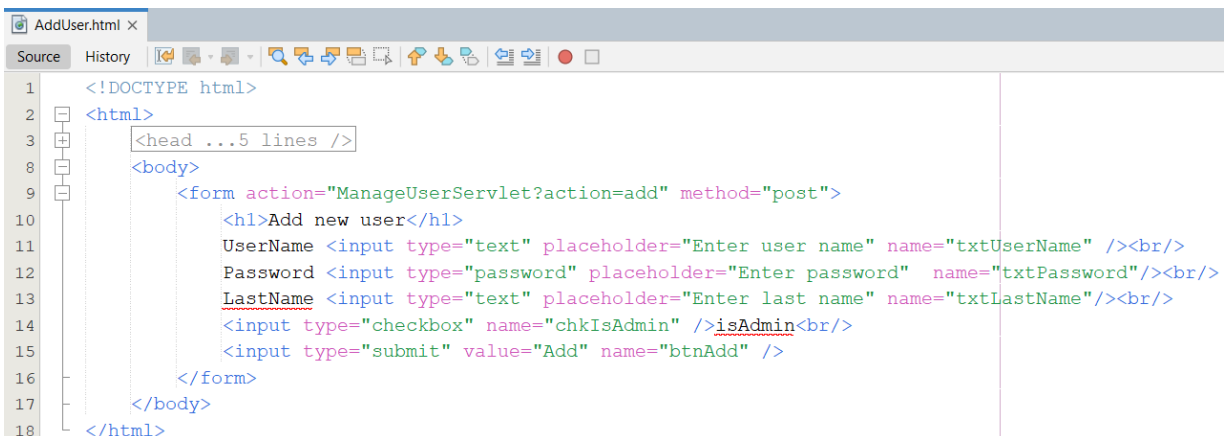
- **DeleteUser.html page**

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Delete User</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <form action="ManageUserServlet?action=delete" method="post">
            <h1>Delete user</h1>
            Enter user name <input type="text" placeholder="Uxxx" name="txtUserName"/><br/>
            <input type="submit" value="Delete" name="btnDelete" />
        </form>
    </body>
</html>
```

- **UpdateUser.html page**

```html
<!DOCTYPE html>
<html>
    <head ...5 lines />
    <body>
        <form action="ManageUserServlet?action=update" method="post">
            <h1>Update user</h1>
            UserName <input type="text" name="txtUserName" value="Uxxx" /><br/>
            Password <input type="password" name="txtPassword" value="" /><br/>
            LastName <input type="text" name="txtLastName" value="" /><br/>
            <input type="checkbox" name="chkIsAdmin" />isAdmin<br/>
            <input type="submit" value="Update" name="btnUpdate" />
        </form>
    </body>
</html>
```

Step 05. Open the **web.xml** in the WEB-INF folder then update as the follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>Login.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Step 06. Right-click on **IntroductionJDBC** project, select Run to run the web app then test all functions.