



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Experiment -8

Student Name: Eklavya

UID:22BCS13380

Branch: BE-CSE

Section/Group:IOT\_639-A

Semester:6<sup>th</sup>

Date of Performance:04/04/2025

Subject Name: Project-Based Learning  
in Java with Lab

Subject Code: 22CSH-359

**7.1.1.Aim:** To develop a servlet that accepts user credentials from an HTML form and displays a personalized welcome message on successful login.

**7.1.2 Objective:** Learn form handling with Servlets  
Understand HTTP request/response handling  
Practice doPost() method

### **7.1.3 Code:**

```
<!DOCTYPE html>
<html>
<head><title>Login</title></head>
<body>
  <form action="LoginServlet" method="post">
    Username: <input type="text" name="username"><br>
    Password: <input type="password" name="password"><br>
    <input type="submit" value="Login">
  </form>
</body>
</html>
```

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

```
public class LoginServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String user = request.getParameter("username");
        String pass = request.getParameter("password");

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        if ("admin".equals(user) && "1234".equals(pass)) {
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
        out.println("<h2>Welcome, " + user + "!</h2>");
    } else {
        out.println("<h2>Login Failed. Invalid credentials.</h2>");
    }
}
}
```

```
<web-app>
<servlet>
  <servlet-name>LoginServlet</servlet-name>
  <servlet-class>LoginServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>LoginServlet</servlet-name>
  <url-pattern>/LoginServlet</url-pattern>
</servlet-mapping>
</web-app>
```

## Output:

- 1) On correct login: Welcome, Eklavya!
- 2) On failure: Login Failed. Invalid credentials.

**7.2.1 Aim:** To build a servlet integrated with JDBC that displays all employees and enables search by employee ID.

**Objective:** 1) Use JDBC with Servlet

2) Fetch and display records

3) Implement search functionality

**7.2.2 Code:**

```
<!DOCTYPE html>
<html>
<head><title>Search Employee</title></head>
<body>
  <form action="EmployeeServlet" method="post">
    Enter Employee ID: <input type="text" name="empId">
    <input type="submit" value="Search">
  </form>
</body>
</html>
```

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
```

```
public class EmployeeServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
```

```
        String empId = request.getParameter("empId");
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
```

```
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
            DriverManager.getConnection("jdbc:mysql://localhost:3306/company", "root", "password");
```

```
            String query = "SELECT * FROM employees WHERE emp_id=?";
            PreparedStatement ps = con.prepareStatement(query);
```

```
ps.setString(1, empId);
ResultSet rs = ps.executeQuery();

if (rs.next()) {
    out.println("<h2>Employee Details</h2>");
    out.println("ID: " + rs.getInt(1) + "<br>");
    out.println("Name: " + rs.getString(2) + "<br>");
    out.println("Department: " + rs.getString(3));
} else {
    out.println("No employee found with ID " + empId);
}

con.close();
} catch (Exception e) {
    out.println("Error: " + e.getMessage());
}
}
```

### 7.2.3 Output:

- 1) Enter an employee ID → Shows details if found.
- 2) Not found → "No employee found with ID X"

**7.3.1 Aim:** To develop a JSP-based student portal that accepts attendance data and saves it to the database using a servlet.

- Objective:**
- 1) Combine JSP for UI and Servlets for logic
  - 2) Perform INSERT using JDBC
  - 3) Build a real-world web flow

**Code:**

```
<%@ page language="java" %>
<html>
<head><title>Student Attendance</title></head>
<body>
  <h2>Mark Attendance</h2>
  <form action="AttendanceServlet" method="post">
    Roll No: <input type="text" name="roll"><br>
    Name: <input type="text" name="name"><br>
    Status: <select name="status">
      <option>Present</option>
      <option>Absent</option>
    </select><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
```

```
public class AttendanceServlet extends HttpServlet {
  protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    String roll = request.getParameter("roll");
    String name = request.getParameter("name");
    String status = request.getParameter("status");

    response.setContentType("text/html");
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
PrintWriter out = response.getWriter();
```

```
try {  
    Class.forName("com.mysql.jdbc.Driver");  
    Connection con =  
DriverManager.getConnection("jdbc:mysql://localhost:3306/student_portal", "root",  
"password");  
  
    String query = "INSERT INTO attendance (roll_no, name, status) VALUES (?, ?,  
?)";  
    PreparedStatement ps = con.prepareStatement(query);  
    ps.setString(1, roll);  
    ps.setString(2, name);  
    ps.setString(3, status);  
  
    int i = ps.executeUpdate();  
    if (i > 0) {  
        out.println("<h3>Attendance marked successfully for " + name + "!</h3>");  
    }  
  
    con.close();  
} catch (Exception e) {  
    out.println("Error: " + e.getMessage());  
}  
}  
}
```

```
CREATE TABLE attendance (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    roll_no VARCHAR(20),  
    name VARCHAR(100),  
    status VARCHAR(10)  
);
```

## OUTPUT

Form submission → "Attendance marked successfully for John!"

And the data is stored in the database.



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Learning Outcome:-

- (i) Learn how to **establish a connection** between a Java application and a MySQL database using **JDBC**.
- (ii) Understand the use of Servlet in java
- (iii) Learn to write sql query and execute it.