

Experiment - 7

Student Name: Kumar Adya

UID: 23BCS80040

Branch: CSE

Section/Group: 636/A

Semester: 6th

Date of Performance: 22.04.25

Subject Name: JAVA

Subject Code: 22CSH-359

1. **Aim:** Develop web applications using Servlets and JSP for user input handling, database integration.
2. **Objective:** Develop dynamic web applications using Servlets and JSP for efficient user input handling and seamless database integration to enhance functionality and user experience.

3. **Implementation/Code:**

□ **Database Table**

```
CREATE DATABASE user_db;  
USE user_db;
```

```
CREATE TABLE users (  
    id INT AUTO_INCREMENT PRIMARY  
    KEY, name VARCHAR(100), email  
    VARCHAR(100)  
);
```

Code

```
import java.io.IOException; import  
java.sql.Connection; import  
java.sql.DriverManager; import  
java.sql.PreparedStatement; import  
javax.servlet.ServletException; import  
javax.servlet.annotation.WebServlet; import  
javax.servlet.http.HttpServlet; import
```

```
javax.servlet.http.HttpServletRequest; import
javax.servlet.http.HttpServletResponse;

@WebServlet("/UserServlet")
public class UserServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {        String name =
    request.getParameter("name");        String email = request.getParameter("email");

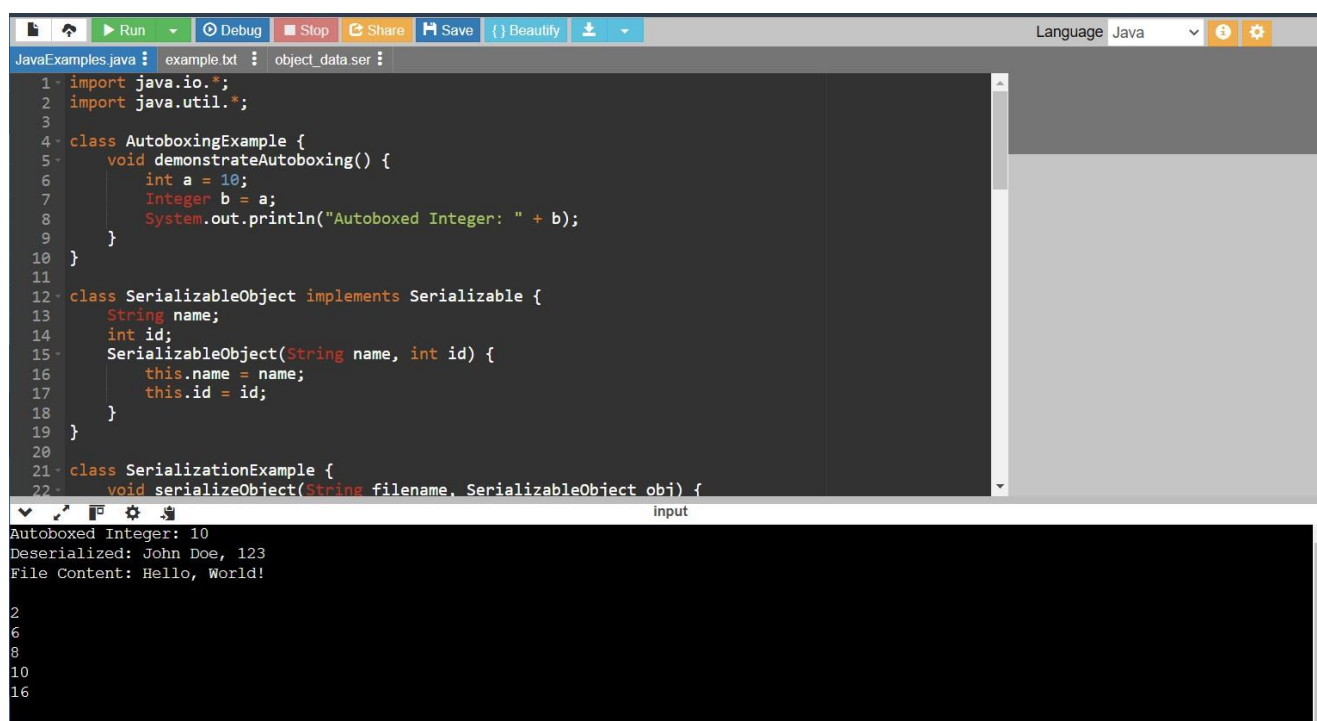
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con =
            DriverManager.getConnection("jdbc:mysql://localhost:3306/user_db", "root",
            "password");
            PreparedStatement ps = con.prepareStatement("INSERT INTO users (name, email)
            VALUES (?, ?)");
            ps.setString(1, name);
            ps.setString(2, email);
            ps.executeUpdate();
            con.close();        } catch
            (Exception e) {
                e.printStackTrace();
            }

            response.sendRedirect("success.jsp");
        }
    }
```

□ JSP form

```
<form action="UserServlet" method="post">
    Name: <input type="text" name="name" required><br>
    Email: <input type="email" name="email" required><br>
    <input type="submit" value="Submit">
</form>
```

4. Output:



The screenshot shows an IDE window with a Java file named 'JavaExamples.java'. The code defines three classes: 'AutoboxingExample', 'SerializableObject', and 'SerializationExample'. The 'AutoboxingExample' class has a method 'demonstrateAutoboxing()' that prints 'Autoboxed Integer: 10'. The 'SerializableObject' class implements 'Serializable' and has fields 'name' and 'id'. The 'SerializationExample' class has a method 'serializeObject()' that prints 'Deserialized: John Doe, 123' and 'File Content: Hello, World!'. The output window at the bottom shows the execution results: 'Autoboxed Integer: 10', 'Deserialized: John Doe, 123', and 'File Content: Hello, World!'. The input field is empty.

```
1- import java.io.*;
2- import java.util.*;
3-
4- class AutoboxingExample {
5-     void demonstrateAutoboxing() {
6-         int a = 10;
7-         Integer b = a;
8-         System.out.println("Autoboxed Integer: " + b);
9-     }
10- }
11-
12- class SerializableObject implements Serializable {
13-     String name;
14-     int id;
15-     SerializableObject(String name, int id) {
16-         this.name = name;
17-         this.id = id;
18-     }
19- }
20-
21- class SerializationExample {
22-     void serializeObject(String filename, SerializableObject obj) {
```

Autoboxed Integer: 10
Deserialized: John Doe, 123
File Content: Hello, World!

5. Learning Outcome:

After implementing these programs, you will:

- ✦ Understand the fundamentals of Servlets and JSP for web development.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

- ✦ Implement user input handling in web applications using Servlets and JSP.
- ✦ Integrate web applications with databases using JDBC.
- ✦ Develop dynamic and interactive web pages using JSP and Servlets.
- ✦ Apply MVC architecture for efficient web application design.
- ✦ Manage session handling and user authentication.
- ✦ Deploy and maintain web applications on a server.