UNIVERSITI MALAYSIA TERENGGANU

FACULTY OF SCIENCE COMPUTER AND MATHEMATICS

CSM3023 (K1)

WEB BASED APPLICATION DEVELOPMENT

LAB 1 : INTRODUCTION TO SERVLET, JSP AND MYSQL DATABASE

Prepared for:

DR MOHAMAD NOR HASSAN

SIR MOHD ARIZAL SHAMSUL BIN MAT RIFIN

Prepared by:

NUR ADILAH AINAA BINTI MOHD NOR

(S67241)

TASK 5

Step 13: You will see the following output on your browser.

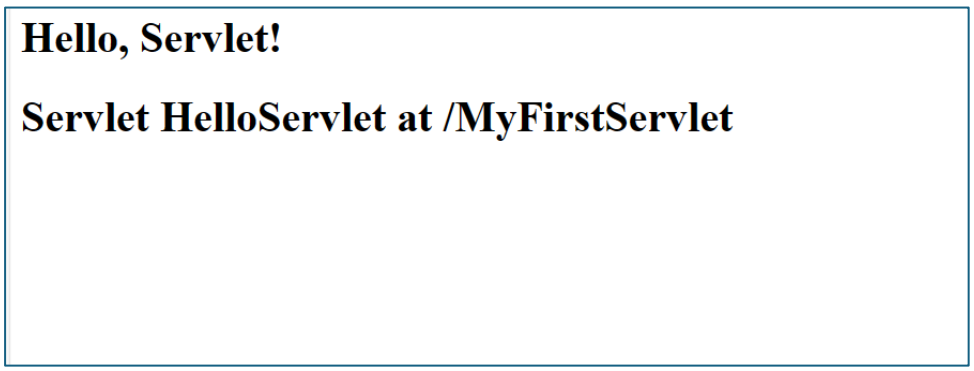
HelloServlet.java

```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 jakarta.annotation.*;

public class HelloServlet extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following sample code. */
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet Saya Yang Pertama</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Hello, Servlet!</h1>");
            out.println("<h1>Servlet HelloServlet at " + request.getContextPath() + "</h1>");
            out.println("</body>");
            out.println("</html>");
        }
    }
```

```java

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {

        /* Step 1 : Set the content type (tell the browser what is teh type of
        response data; e.g text/html, text/plain. In our case, we will respons
        with html data.
        */
        response.setContentType("text/html");

        /*Step 2: Create tje PrintWriter object. We name it as 'out'
        */
        PrintWriter out = response.getWriter();

        /*Step 3: Read GET parameter sent by the user through the web browser*/
        String name = request.getParameter("name");

        /*Additional:
        if no value for parameter "name", cll processRequest method.*/
        if(name == null)
            processRequest(request, response);

        /*Step 4: Generate content for our HTML response. Print the name*/
        out.println("<html><body>");

```

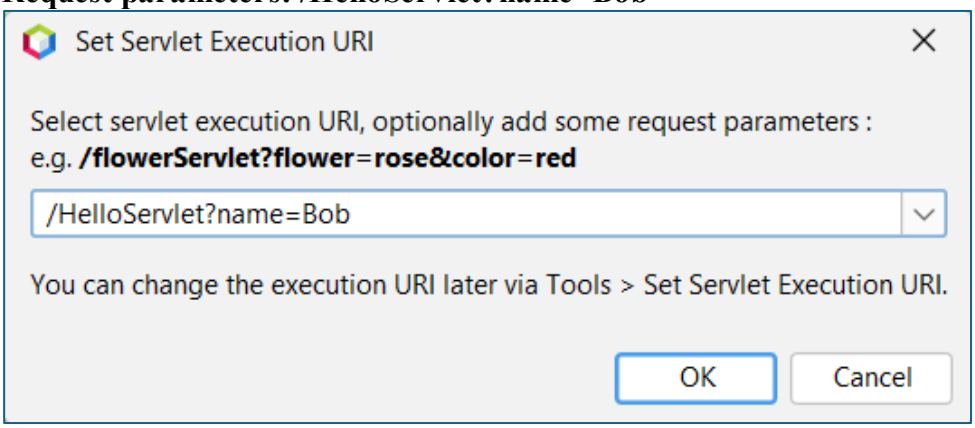
```
        out.println("Hello, " + name + "!"+"<br>");
        out.println("Apa khabar?"+ "<br>");
        out.println("Waktu dan tarikh di Server ialah " + new java.util.Date());
        out.println("</html></body>");
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>
```

Output:



Step 13: After finish, right click on HelloServlet.java and click Run. As you have seen previously, a dialogue box shows. This time, we will supply a value Bob to the parameter name. Then, click the OK button.

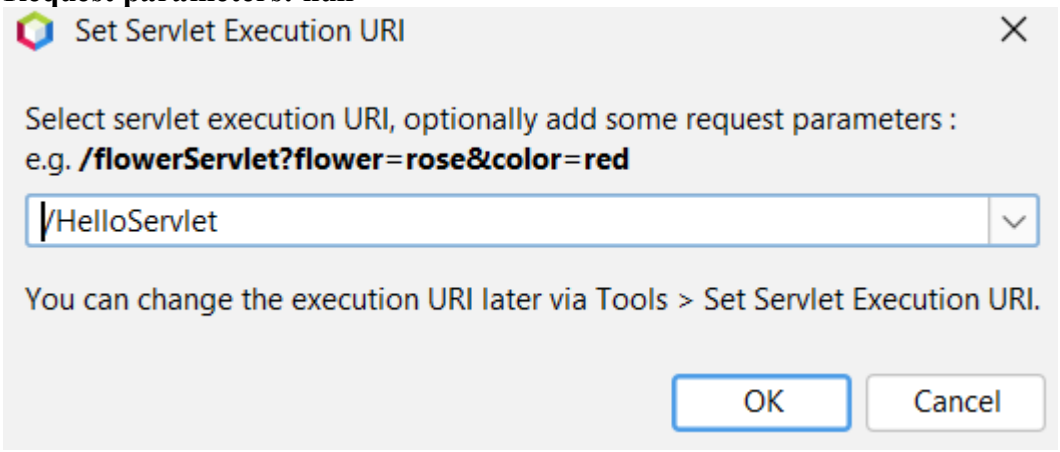
A) Request parameters: /HelloServlet?name=Bob



Output:

Hello, Bob!
Apa khabar?
Waktu dan tarikh di Server ialah Sat Mar 30 22:21:27 MYT 2024

B) Request parameters: null



Hello, null!
Apa khabar?
Waktu dan tarikh di Server ialah Sat Mar 30 22:27:31 MYT 2024

Step 18: .You can upgrade your code in HelloServlet.java by putting the following codes into it. By doing this, if no value supplied to the parameter name, the request will be passed to processRequest() method, and this will avoid from the null value appears on the browser.

Step 19: So, if you rerun the file and without supplying any parameter, you will see the output as follows:

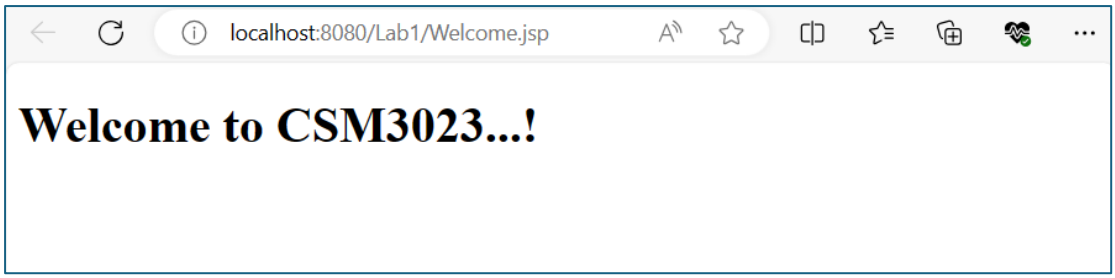
Output:

Question : It is the same output as can be seen in Step 18: why?

Answer : This is because that we have updated the code. If null parameter is supplied to the request, then the output will be displayed as what is written in the processRequest method.

TASK 6

Output:



Source Code:

```jsp
<%--
    Document   : Welcome
    Created on : 30 Mar 2024, 12:41:03 pm
    Author     : User
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>CSM3023 - Web Programming 2</title>
    </head>
    <body>
        <h1>Welcome to CSM3023...!</h1>
    </body>
</html>
```

Reflection:

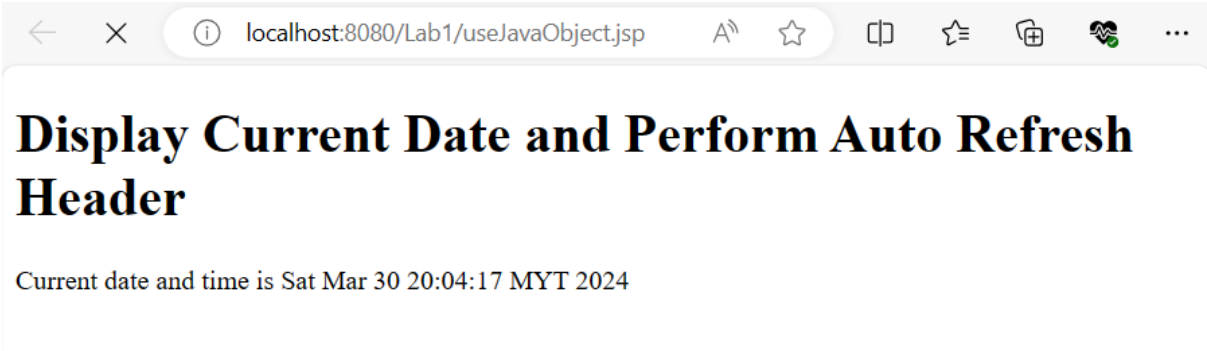
1. **What have you learned from this exercise?**
   - I have successfully learned on how to create a simple plain JSP program.

2. **Explain the general concept of how the JSP's file work?**
   - Web browser sends request to the web server
   - The web server recognizes the request as JSP page
   - The JSP engine fetch the JSP page and convert to servlet.
   - The JSP engine converts the servlet into executable class and send the request to servlet engine.

3. **Based on your observation of previous task (3 and 4), what are the differences you can find between servlet and JSP?**
   - Servlet is a server-side Java program module that process and respons to client request by implementing the servlet interface. While JSP is a web scripting language that helps developers to create dynamic web pages.
   - Servlet's man focus is on information processing. While JSP's main focus is in displaying information

TASK 7

Output:



Source Code:

```
<%--
    Document   : useJavaObject
    Created on : 30 Mar 2024, 12:46:41 pm
    Author     : User
--%>

<%@page import="java.util.Date"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Using Java's Object in JSP Page</title>
    </head>
    <body>
        <h1>Display Current Date and Perform Auto Refresh Header</h1>
        <%
        Date todayDate = new Date();
        out.print("<p>Current date and time is " +todayDate.toString() + "</p>");
        %>
        <%
        response.setIntHeader("Refresh", 5);
        %>
    </body>
</html>
```

Reflection:

1. **What have you learnt from this exercise?**

   - I have successfully learned on how to use Java's object in JSP page by trying to display current date and perform auto refresh header

2. **What is Java Scriplet?**

   - A JSP scriptlet is a container for any code fragment that is valid for the page's scripting language.

3. **How to use Java code in your JSP's page?**

   - Create a JSP page that contains the <%code fragment%> script. It can contain any number of JAVA language statements, variable or method declaration, or expressions that are valid in the page scripting language.

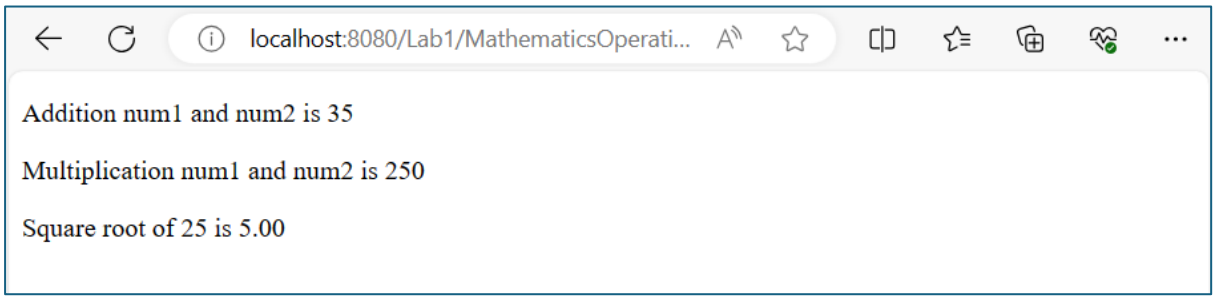
   - Keep any html tags in the page outside the scriptlet.

   - Use the import attribute inside the %@page...% directive to define any packages for use inn the page, just like the Java import statement does for java classes.

TASK 8

Step 10: Run the AttributeIsSet.jsp file, and you should get the interface as below:

Output:



Source code:

AttributeIsSet.jsp:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.math.*"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
        <title>Implicit JSP</title>
    </head>
    <body>
        <% session.setAttribute("user", "Adilah Ainaa");%>
        <a href = "GetAttribute.jsp"> Click here to get user name </a>
        <br>
        <a href = "MathematicsOperations.jsp"> Result of mathematics operations </a>
    </body>
</html>
```

GetAttribute.jsp:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Implicit JSP</title>
    </head>
    <body>
        <%
            String name = (String) session.getAttribute("user");
            out.println("User Name is: " + name);
        %>
    </body>
</html>
```

MathematicsOperations.jsp:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
        <title>JSP Page</title>
    </head>
    <body>
        <%
            int num1 = 25;
            int num2 = 10;
            int addition_output;
            int multiply_output;
            double squareroot = 0.00;

            java.util.Formatter myFormat = new java.util.Formatter();

            addition_output = num1 + num2;
            multiply_output = num1 * num2;

            squareroot = (double)(Math.sqrt(num1));

            out.print("<p>Addition num1 and num2 is " + addition_output + "</p>");
            out.print("<p>Multiplication num1 and num2 is " + multiply_output + "</p>");

            out.print("<p></p>");
            out.print("<p>Square root of " + num1 + " is " + myFormat.format("%.2f",
squareroot)+"</p>");
        %>
    </body>
</html>
```

Reflection:

1. **How do you want to submit specific information from one form to next form?**

    A. Use session.setAttribute(“”, “”) to one form & session.getAttribute(“field_name”) to another form.

B. Link the two form by using <a href="File_Name.jsp"> tag.

2. **What happened if the field name you specify in request.getParameter ("field_name") in the second page is different from the field name you defined in the first page?**

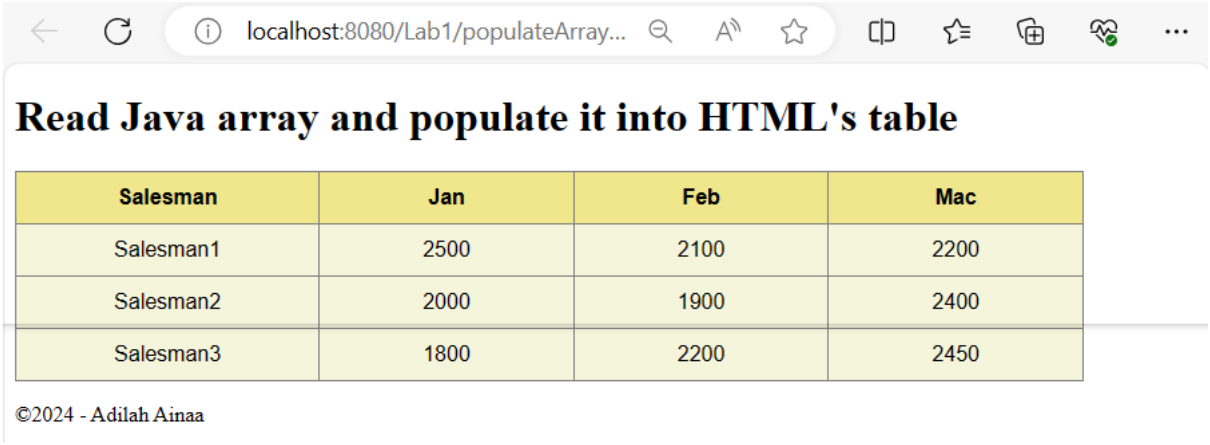
   - An Exception report will be displayed.

     "An error occurred at line:[17] in the jsp file: [/GetAttribute.jsp]

     Cannot cast form Object to int"

TASK 9

Output:



Source code:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Populate Array Into Table</title>
        <style>
            table, th, td {
                font-family: Arial, Helvetica, sans-serif;
                border: 1px solid grey;
                border-collapse: collapse;
                height: 5px;
                width: 800px;
                text-align: center;
                padding: 10px;
            }
            thead {
                background-color: khaki;
            }
            td {
                background-color: beige;
            }
        </style>
    </head>
    <body>
        <h1>Read Java array and populate it into HTML's table</h1>

        <table>
            <thead>
                <%
                    String[][] headers =
                        {
                            {"Salesman", "Jan", "Feb", "Mac"}
                        };

```

```
                    for (String[] head : headers) {
                        out.print("<tr>");
                        for (String salesmanData : head) {
                            out.print("<th>" + salesmanData + "</th>");
                        }
                        out.print("</tr>");
                    }
                %>
            </thead>
            <tbody>
                <%
                    String[][] salesmen =
                        {
                            {"Salesman1", "2500", "2100", "2200"},
                            {"Salesman2", "2000", "1900", "2400"},
                            {"Salesman3", "1800", "2200", "2450"}
                        };

                    for (String[] salesman : salesmen) {
                        out.print("<tr>");
                        for (String salesmanData : salesman) {
                            out.print("<td>" + salesmanData + "</td>");
                        }
                        out.print("</tr>");
                    }
                %>
            </tbody>
        </table>
        <p>&#169;2024 - Adilah Ainaa </p>
    </body>
</html>
```

Reflection:

1. **Write a simple syntax to declare 2D java array**

   - data_type[][] array_name = new data_type [x][y];

2. **Define a sequence of steps on how you accomplish Task 7**

   - Create JSP file
   - Write anything in the <h1> tag
   - Write java code to call built in method Date
   - Write java code to perform auto refresh header.

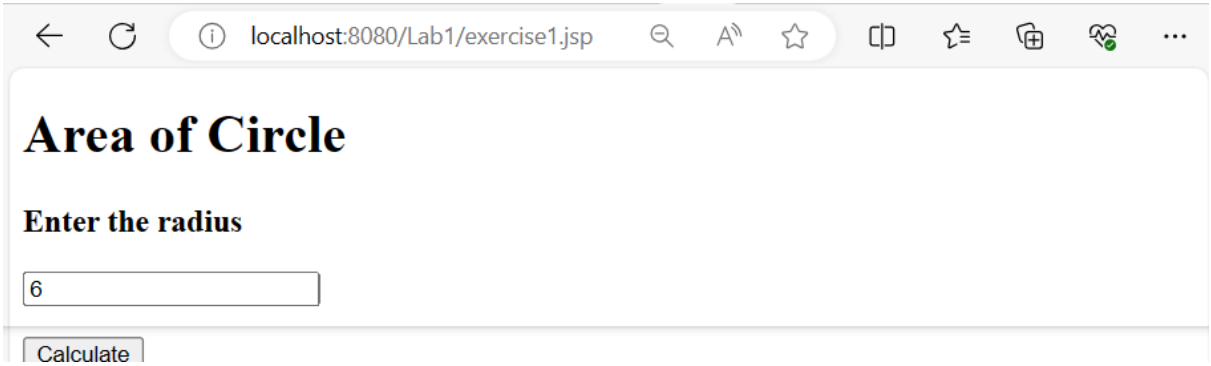
3. **What is the difference between HTML page and JSP page?**

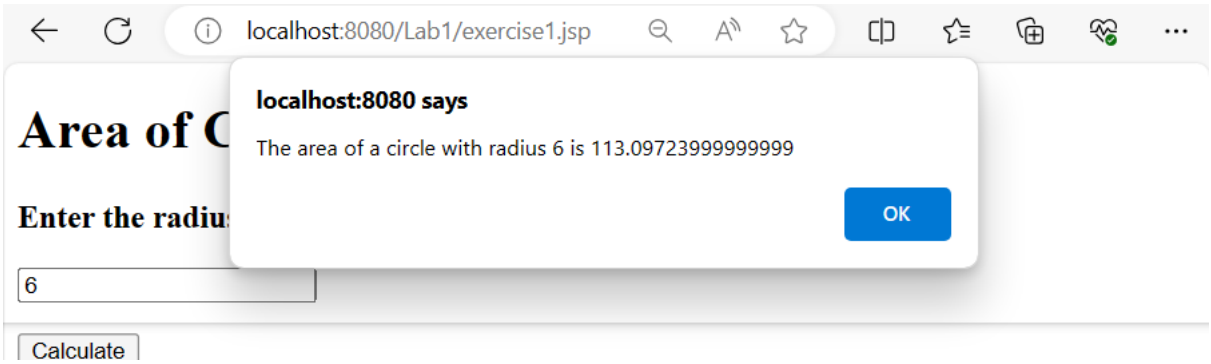
   - HTML page is a static web page while JSP page is a dynamic web page

EXERCISES

1. Exercise 1

- **Output:**



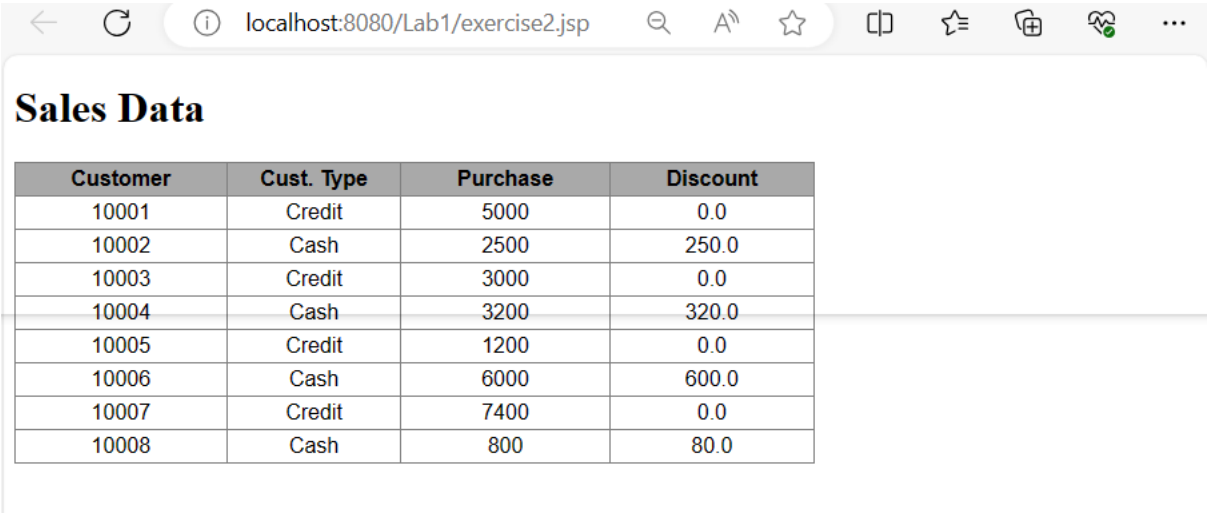


- **Source Code:**

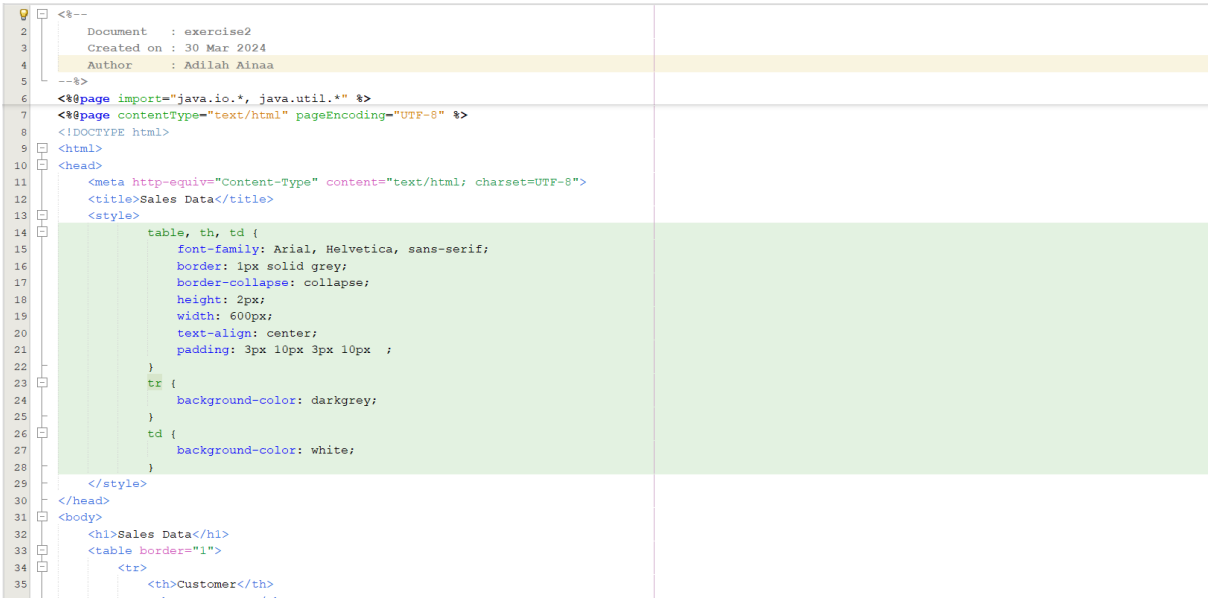
```jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Area of Circle</title>
</head>
<body>
    <h1>Area of Circle</h1>
    <%
        double circleArea = 0.0;
        double pi = 3.14159;
        double radius = 0.0;
        String input = request.getParameter("radius");
        if (input != null && !input.isEmpty()) {
            radius = Double.parseDouble(input);
            circleArea = pi * radius * radius;
        }
    %>

    <h3>Enter the radius</h3>
    <input id="myInput" type="number" min="0" step="0.01" value=""> <br><br>
    <button id="myBtn" onclick="calculateArea()">Calculate</button>

    <script>
        function calculateArea() {
            var radius = parseFloat(document.getElementById("myInput").value);
            if (!isNaN(radius) && radius > 0) {
                var pi = 3.14159;
                var circleArea = pi * radius * radius;
                alert("The area of a circle with radius " + radius + " is " + circleArea);
            } else {
                alert("Please enter a positive and valid radius.");
            }
        }
    </script>
```

2. Exercise 2

- **Output:**



- **Source Code:**



```
<%--
    Document   : exercise2
    Created on : 30 Mar 2024
    Author     : Adilah Ainaa
--%>
<%@page import="java.io.*, java.util.*" %>
<%@page contentType="text/html" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Sales Data</title>
    <style>
            table, th, td {
                font-family: Arial, Helvetica, sans-serif;
                border: 1px solid grey;
                border-collapse: collapse;
                height: 2px;
                width: 600px;
                text-align: center;
                padding: 3px 10px 3px 10px  ;
            }
            tr {
                background-color: darkgrey;
            }
            td {
                background-color: white;
            }
    </style>
</head>
<body>
    <h1>Sales Data</h1>
    <table border="1">
        <tr>
            <th>Customer</th>
```

```
            <th>Cust. Type</th>
            <th>Purchase</th>
            <th>Discount</th>
        </tr>
        <%
            try {
                String csvFile = "C:/CSF3023 - PROJECT/Lab1/Sales.csv";
                BufferedReader br = new BufferedReader(new FileReader(csvFile));
                String line;

                while ((line = br.readLine()) != null) {
                    String[] data = line.split(";");
                    String customer = data[0];
                    String custType = data[1];
                    double purchase = Double.parseDouble(data[2]);
                    int purchaseInt = (int) purchase;
                    double discount = 0.0;

                    if (custType.equals("Cash")) {
                        discount = purchaseInt * 0.10;
                    }

                    out.println("<tr>");
                    out.println("<td>" + customer + "</td>");
                    out.println("<td>" + custType + "</td>");
                    out.println("<td>" + purchaseInt + "</td>");
                    out.println("<td>" + discount + "</td>");
                    out.println("</tr>");
                }

                br.close();
            } catch (FileNotFoundException e) {
                out.println("Sales.csv file not found.");
            } catch (IOException e) {
                out.println("Error reading Sales.csv file.");
            } catch (NumberFormatException e) {
                out.println("Error parsing numeric data.");
            }
        %>
    </table>
</body>
</html>
```