



# **UNIVERSITI MALAYSIA TERENGGANU**

**Faculty of Computer Sciences and Mathematics**

**Web Based Application Development**

**CSM3023**

**Lab Report 1**

**Prepared for:**

**Sir. Mohd Arizal Shamsil bin Mat Rifin**

**Prepared by:**

**LUTFIL HAZIQ BIN ADNAN(S67911)**

**27<sup>th</sup> March 2024**

**Bachelor of Computer Science (Mobile Computing) with Honors**

**Semester II 2023/2024**

## Task 5 : Linking NetBeans to Apache Tomcat and Writing a Simple Java Servlet

- Objective: To write a simple Java Servlet.
- Code:

```
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;

/**
 *
 * @author LUTFIL HAZIQ BIN ADNAN
 */
public class HelloServlet extends HttpServlet {

    /**
     * Processes requests for both HTTP GET and
     * POST
     * 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("<h2>Servlet HelloServlet at " +
                request.getContextPath() + "</h2>");
            out.println("</body>");
            out.println("</html>");
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet
    methods. Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP GET 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 context type (tell the browser what is the
            type of the response data; e.g text/html, text/plain). In our case, we
            will responds with html data.
        */
        response.setContentType("text/html");

        /*
            Step 2: Create the 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", call
            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?<hr>");
        out.println("Waktu dan tarikh di Server ialah " + new
            java.util.Date());
        out.println("</body></html>");
    }

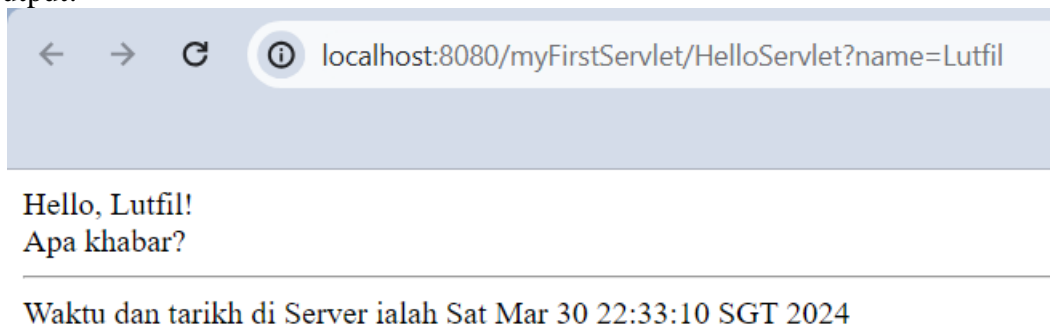
    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest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



- Reflection:

It is the same output as can be seen in Step 18. Why?

It produces the same result since the same method, `processRequest()`, was used.

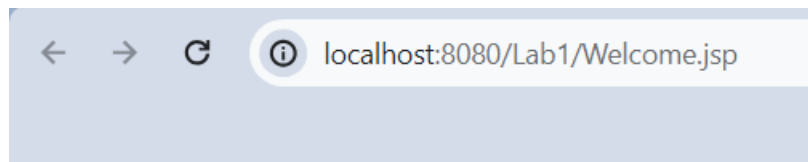
## Task 6 : Writing a Simple JSP Program

- Objective: Writing a simple plain JSP program.
- Code:

```
<%--
    Document    : welcome
    Created on   : 29 Mar 2024, 3:34:29 pm
    Author      : LUTFIL HAZIQ BIN ADNAN

<%@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 Based Application Development</title>
    </head>
    <body>
        <h1>welcome to CSM3023...!</h1>
    </body>
</html>
```

- Output:



**Welcome to CSM3023...!**

- Reflection:

1. What have you learned from this exercise?

I learnt how to set up a JSP program in NetBeans using Apache Tomcat server.

2. Explain the general concept of how the JSP's file work?

JSP files combine HTML with special tags to create dynamic web content. When requested, the server translates JSP into a Java servlet, allowing interaction with databases or other server-side operations. The servlet generates HTML pages dynamically, enabling personalized content based on user requests or server data.

3. Based on your observation of the previous tasks (Task 3 and Task 4), what are the differences you can find between servlet and JSP?

Servlets handle backend tasks like data processing and database communication, written in Java. JSPs focus on presentation, blending HTML with JSP tags for easy development. They are converted into servlets before execution. While servlets handle application logic, JSPs control how users view content.

## Task 7 : Use Java Reference Datatype/Class Wrapper in JSP

- Objective: Display the current date, perform auto refresh header in JSP's page.
- Code:

```
<!--
    Document    : useJavaObject
    Created on  : 29 Mar 2024, 3:34:02 pm
    Author      : LUTFIL HAZIQ BIN ADNAN
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>

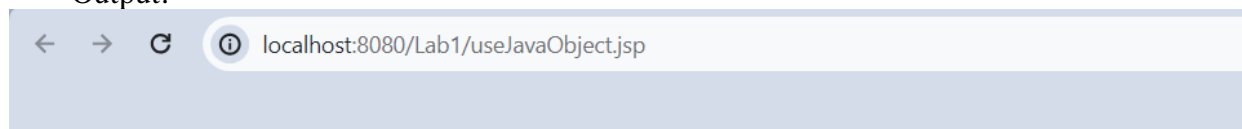
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
        <title>Using Java's object in JSP page</title>
    </head>
    <body>
        <h1>Display current date and perform auto refresh header</h1>

        <%
            Date todayDate = new Date();
            out.print("<p>Current date and time is " +
todayDate.toString() + "</p>");

            // Set refresh, autoloading time as seconds
            response.setIntHeader("Refresh", 5);
        %>

    </body>
</html>
```

- Output:



## Display current date and perform auto refresh header

Current date and time is Sat Mar 30 22:35:46 SGT 2024

- Reflection:

1. What have you learnt from this exercise?

From this exercise I learnt how to use the `<% %>` symbol which is scriptlet tag used to embed Java code within the HTML content of a JSP file. We can insert Java code in HTML by using scriptlet tag.

2. What is Java Scriptlet?

A Java Scriptlet, enclosed within `<% %>`, embeds dynamic Java logic directly into JSP pages. It allows tasks like variable assignment, database access, and calculations within the JSP file. While useful, excessive use can complicate code and pose security risks.

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

In JSP, scriptlets (`<% %>`) execute Java code for operations like data processing and decision making. Expressions within brackets evaluate and output Java directly into HTML responses. Declarations (`<%! %>`) define variables, methods, and classes for broader usage in the JSP page.



## Task 8 : Using JSP Implicit object in JSP page

- Objective: Using Session object, perform simple Mathematics operations in JSP's page.
- Code:

```
<%--
    Document    : AttributeIsSet
    Created on  : 29 Mar 2024, 3:43:18 am
    Author     : LUTFIL HAZIQ BIN ADNAN
--%>

<%@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>

        <% session.setAttribute("user", "LUTFIL HAZIQ "); %>
        <a href="GetAttribute.jsp">Click here to get user name</a>
        <br>
        <a href="MathematicsOperations.jsp">Result of mathematics
operations</a>

    </body>
</html>
```

```
<%--
    Document    : GetAttribute
    Created on  : 29 Mar 2024, 7:29:15 am
    Author     : LUTFIL HAZIQ BIN ADNAN
--%>

<%@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>
```

```
<%--
    Document    : MathematicsOperations
    Created on   : 29 Mar 2024, 9:33:10 am
    Author      : LUTFIL HAZIQ BIN ADNAN
--%>

<%@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>

        <%
            int num1 = 25,
                num2 = 10,
                addition_output,
                multiply_output;
            double squareroot = 0.00;

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

            // Perform basic arithmetics operations
            addition_output = num1 + num2;
            multiply_output = num1 * num2;

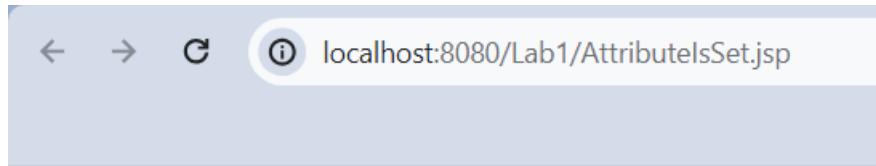
            // Find square root for variable num1
            squareroot = (double)(Math.sqrt(num1));

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

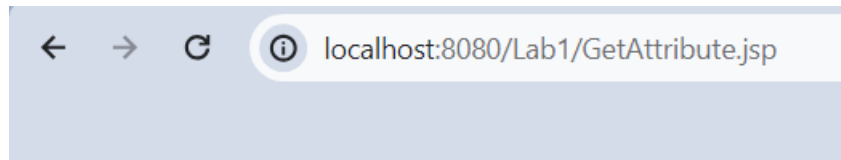
        %>

    </body>
</html>
```

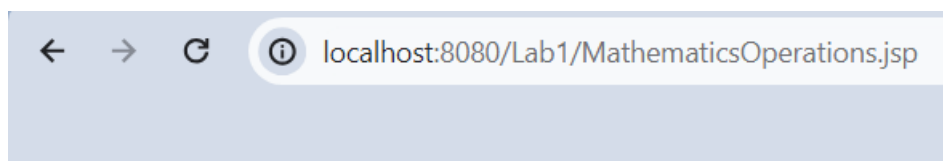
- Output:



[Click here to get user name](#)  
[Result of mathematics operations](#)



User name is: LUTFIL HAZIQ



Addition num1 and num2 is 35.

Multiplication num1 and num2 is 250.

Square root of 25 is 5.00.

- Reflection:

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

By using href attributes. It is because the href property indicates the URL of the linked page or the page's next destination, similar to a hyperlink.

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?

It will show null.

## Task 9 : Populate Array values into HTML's Table

- Objective: Read Java array and populate it into HTML's table.
- Code:

```
<%--
    Document    : populateArray
    Created on  : 29 Mar 2024, 7:48:21 am
    Author     : LUTFIL HAZIQ BIN ADNAN
--%>

<%@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</title>
        <style>
            table, th, td {
                border: 1px solid black;
                border-collapse: collapse;
                padding: 8px 30px;
                text-align: center;
            }

            th {
                background-color: yellow;
            }
        </style>
    </head>
    <body>

        <h1>Read Java array and populate it into HTML's table</h1>

        <%
            String[][] sales = {
                {"Salesman", "Jan", "Feb", "Mar"},
                {"Salesman 1", "2500", "2100", "2200"},
                {"Salesman 2", "2000", "1900", "2400"},
                {"Salesman 3", "1800", "2200", "2450"}
            };
        %>
```

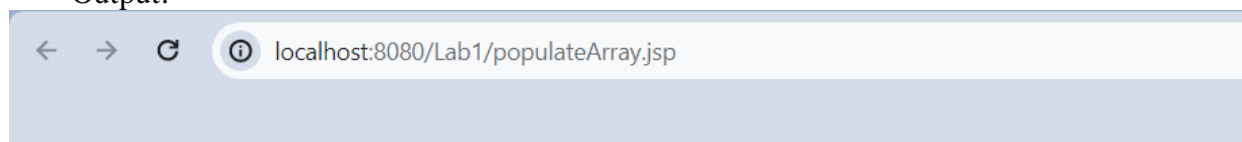
```

<table border="1">
  <thead>
    <tr>
      <% for (int j = 0; j < sales[0].length; j++) { %>
        <th><%= sales[0][j] %></th>
      <% } %>
    </tr>
  </thead>
  <tbody>
    <% for (int i = 1; i < sales.length; i++) { %>
      <tr>
        <% for (int j = 0; j < sales[i].length; j++) {
%>
          <td><%= sales[i][j] %></td>
        <% } %>
      </tr>
    <% } %>
  </tbody>
</table>

</body>
</html>

```

- Output:



## Read Java array and populate it into HTML's table

| Salesman   | Jan  | Feb  | Mar  |
|------------|------|------|------|
| Salesman 1 | 2500 | 2100 | 2200 |
| Salesman 2 | 2000 | 1900 | 2400 |
| Salesman 3 | 1800 | 2200 | 2450 |

- Reflection:

1. Write a sample syntax to declare 2D Java array.

```
datatype[][] arrayName = new dataType[row][columns];
```

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

- Create a new JSP file and name it as useJavaObject.
- Import java.util.Date.
- Create Date object and display the current date and time.
- Use setInterval() to trigger refresh function and set the autoloading time to 5 seconds.
- Compile and run the program.

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

HTML is for static web pages, ideal for displaying regular content quickly. JSP (JavaServer Pages) allow dynamic content with HTML and Java. Unlike HTML, JSP embeds Java within HTML for dynamic generation. JSP undergo runtime processing to create dynamic HTML based on calculations. This facilitates interactive, data-driven web apps fetching data from various sources. JSP also simplifies interaction with Java servlets and backend technologies, aiding in better separation of presentation and business logic.

## Exercise 1

- Objective: To convert a Java program into Web-Based application.
- Code:

```
<%--
    Document    : getRadius
    Created on   : 29 Mar 2024, 8:43:51 am
    Author      : LUTFIL HAZIQ BIN ADNAN
--%>

<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Calculate Circle Area</title>
    </head>
    <body>
        <h1>Calculate Circle Area</h1>
        <form method="post" action="calculateArea.jsp">
            <label for="radius">Enter the radius:</label>
            <input type="text" id="radius" name="radius" pattern="[0-9]+" title="Please enter a number" required>
            <button type="submit">Calculate</button>
        </form>
    </body>
</html>
```

```
<%--
    Document    : calculateArea
    Created on   : 29 Mar 2024, 8:47:27 am
    Author      : LUTFIL HAZIQ BIN ADNAN
--%>

<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%@ page import="java.util.Scanner" %>
<%@ page import="java.text.DecimalFormat" %>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Area Result</title>
    </head>
    <body>
        <h1>Area Result</h1>
        <%-- Retrieve the radius value from the form --%>
        <% String radiusStr = request.getParameter("radius"); %>

        <%-- Check if the radius value is provided and is a valid
number --%>
```

```

<% if (radiusStr != null && radiusStr.matches("\\d+")) { %>
    <!-- Convert the radius string to an integer --%>
    <% int radius = Integer.parseInt(radiusStr); %>

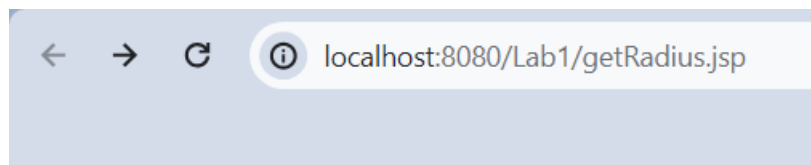
    <!-- Calculate the area --%>
    <% double area = Math.PI * radius * radius; %>

    <!-- Format the area value --%>
    <% DecimalFormat df = new DecimalFormat("#.##"); %>

    <!-- Display the result --%>
    <p>The area of the circle with radius <%= radius %> is <%=
df.format(area) %>.</p>
    <% } else { %>
        <!-- Display an error message if the radius value is
missing or not a valid number --%>
        <p>Please enter a valid number for the radius.</p>
    <% } %>
</body>
</html>

```

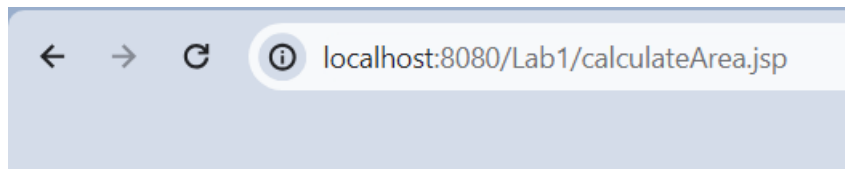
- Output:



## Calculate Circle Area

Enter the radius:





## Area Result

The area of the circle with radius 25 is 1963.5.

### Exercise 2

- Objective: To create a simple JSP's page to read CSV file.
- Code:

```
<!--
  Document    : readCSVFile
  Created on  : 29 Mar 2024, 9:17:24 am
  Author      : LUTFIL HAZIQ BIN ADNAN
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ page import="java.io.*, java.util.*" %>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
    <title>Read CSV File</title>
    <style>
      table, th, td {
        border: 1px solid black;
        border-collapse: collapse;
        padding: 8px 30px;
        text-align: center;
      }

      th {
        background-color: lightgray;
      }
    </style>
  </head>
  <body>
    <table border="1">
      <tr>
        <th>File Name</th>
        <th>File Content</th>
      </tr>
      <tr>
        <td>data.csv</td>
        <td>1,2,3,4,5</td>
      </tr>
    </table>
  </body>
</html>
```

```

</style>
</head>
<body>

<h1>Read CSV files and populate it into HTML's table</h1>

<%
    // Path to the CSV file
    String filePath =
"C:/Users/ACER/Downloads/Lab1/Sales.csv";

    // List to hold the data from the CSV file
    List<String[]> csvData = new ArrayList<>();

    // Read the CSV file and populate the list
    try {
        BufferedReader br = new BufferedReader(new
FileReader(filePath));
        String line;
        while ((line = br.readLine()) != null) {
            String[] parts = line.split(",");
            csvData.add(parts);
        }
        br.close();
    } catch (IOException e) {
        out.println("Error reading CSV file: " +
e.getMessage());
    }
%>

<table>
<thead>
<tr>
<th>Customer</th>
<th>Cust. Type</th>
<th>Purchase</th>
<th>Discount</th>
</tr>
</thead>
<tbody>
<% for (int i = 0; i < csvData.size(); i++) { %>
    <tr>
        <%
            String[] rowData = csvData.get(i);
            String customer = rowData[0];
            String custType = rowData[1];
            int purchase =
Integer.parseInt(rowData[2]);
            double discount = 0.0;
            if (custType.equalsIgnoreCase("Cash")) {
                discount = purchase * 0.1; // 10%
discount for cash customers
            }
        %>
        <td><%= customer %></td>
        <td><%= custType %></td>
        <td><%= purchase %></td>
        <td><%= String.format("%.2f", discount)
%></td>

```

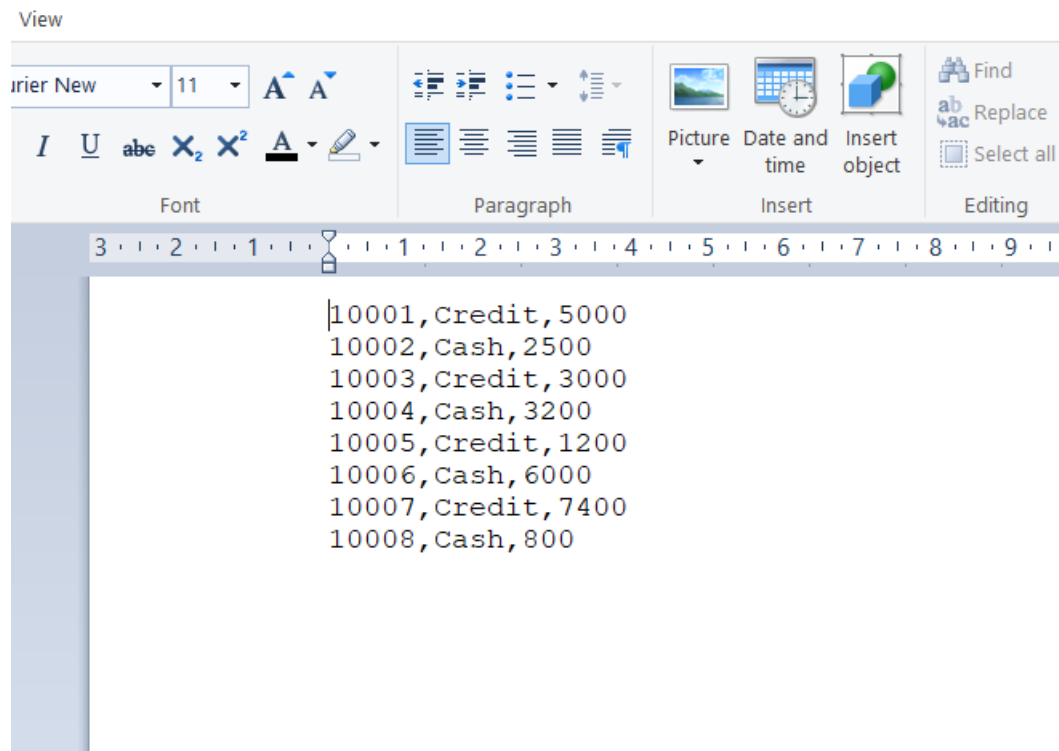
```

        </tr>
      <% } %>
    </tbody>
  </table>

</body>
</html>

```

Sales - wordpad



*Sales.csv*

- Output:

## Read CSV files and populate it into HTML's table

| Customer | Cust. Type | Purchase | Discount |
|----------|------------|----------|----------|
| 10001    | Credit     | 5000     | 0.00     |
| 10002    | Cash       | 2500     | 250.00   |
| 10003    | Credit     | 3000     | 0.00     |
| 10004    | Cash       | 3200     | 320.00   |
| 10005    | Credit     | 1200     | 0.00     |
| 10006    | Cash       | 6000     | 600.00   |
| 10007    | Credit     | 7400     | 0.00     |
| 10008    | Cash       | 800      | 80.00    |