Week 2

JSP: Scriplet, Page Directive & Include Directive

Web Programming 2

Name: NURHASLINDA BINTI BAHARUDDIN

Matric #: S67383

Semester:4

Lab: Lab 3
Demonstrator: SIR ARIZAL

Lecturers
PUSAT PENGAJIAN INFORMATIK DAN MATEMATIK GUNAAN (PPIMG), UNIVERSITI MALAYSIA TERENGGANU (UMT)

Revision History

Revision Date	Previous Revision Date	Summary of Changes	Changes Marked
		First Issue	Mohamad Nor Hassan
		Second Issue	Dr Rabiei Mamat Dr Faizah Aplop Dr Fouad Ts Dr Rosmayati Mohemad Fakhrul Adli Mohd Zaki
21/02/2019		Addition of Revision History, Table of Contents, Formatting Cover Page	Fakhrul Adli Mohd Zaki

Table of Contents

Arahan:

Manual makmal ini adalah untuk kegunaan pelajar-pelajar Pusat Pengajian Informatik dan Matematik Gunaan (PPIMG), Universiti Malaysia Terengganu (UMT) sahaja. Tidak dibenarkan mencetak dan mengedar manual ini tanpa kebenaran rasmi daripada penulis.

Sila ikuti langkah demi langkah sebagaimana yang dinyatakan di dalam manual. Tandakan (√) setiap langkah yang telah selesai dibuat dan tuliskesimpulan bagi setiap aktiviti yang telah selesai dijalankan.

Instruction:

This laboratory manual is for use by the students of the School of Informatics and Applied Mathematics (PPIMG), Universiti Malaysia Terengganu (UMT)only. It is not permissible to print and distribute this manual without the official authorisation of the author.

Please follow step by step as described in the manual. Tick (√) each step completed and write the conclusions for each completed activity.

Task 1: Passing Data from Main JSP's Page to Other JSP's Page

Objective : To demonstrate the use of *request.getParameter (“fieldName”)* for passing input from one JSP'page to another JSP's page.

Problem : i. Create a page *memberRegister.jsp*.

Description ii. Page *memberRegister.jsp* consists of two (2) inputs;

- IC No (Must be in pre-formatted XXXXXXXXXX)
- Name

3. In *memberRegister.jsp*, include two (2) buttons;

   Submit and *Cancel* button.

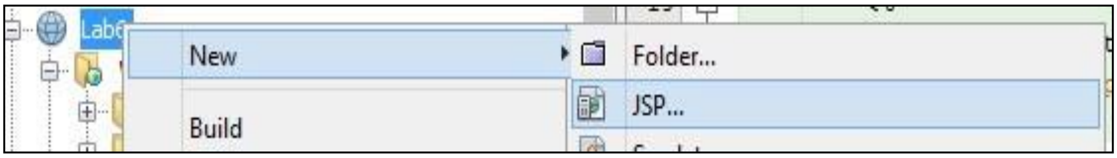
4. Create a page *memberProcessing.jsp*.

5. When user click *Submit* button, process the request and display the input key-in in *memberProcessing.jsp* page.

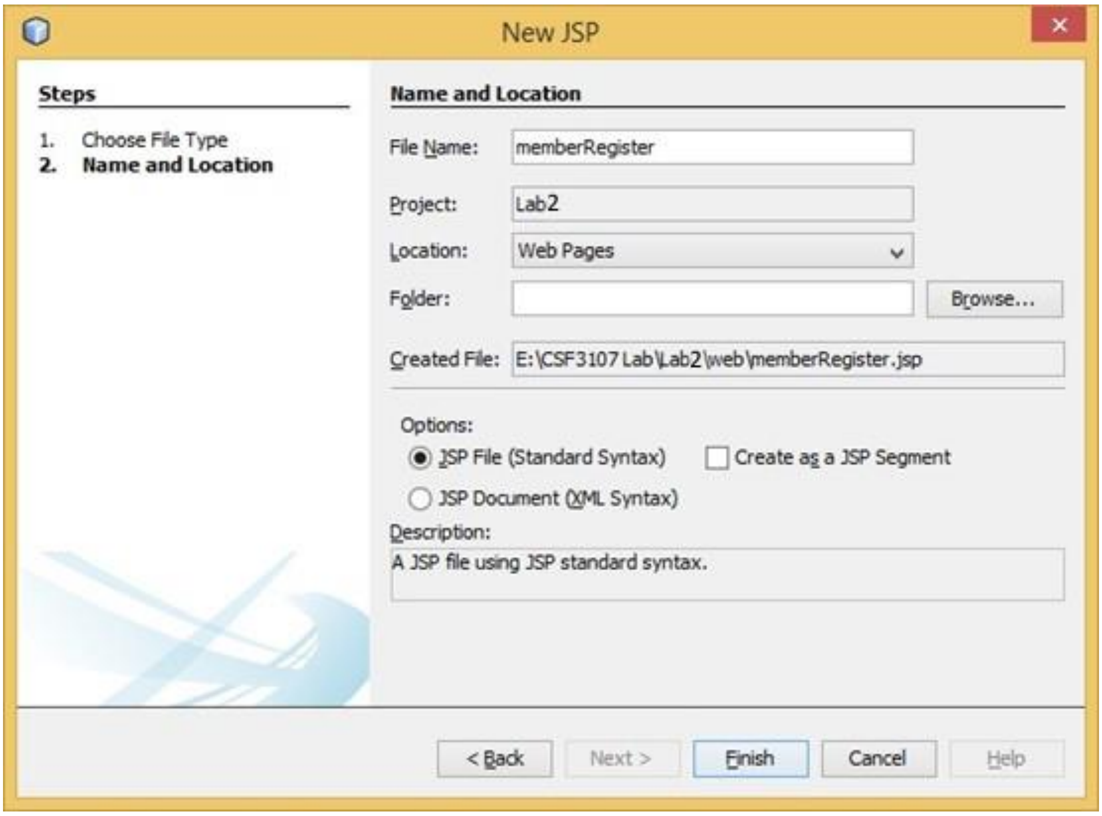
Estimated time : 30 minutes

1. Create new Project namely *Lab2*.

2. To create a JSP's page, right click *Lab2* -> *New* -> *JSP*.



3. Key-in File Name: *memberRegister*.



File Name: memberRegister

Project: Lab2

Location: Web Pages

Folder:

Created File: E:\CSF3107 Lab\Lab2\web\memberRegister.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

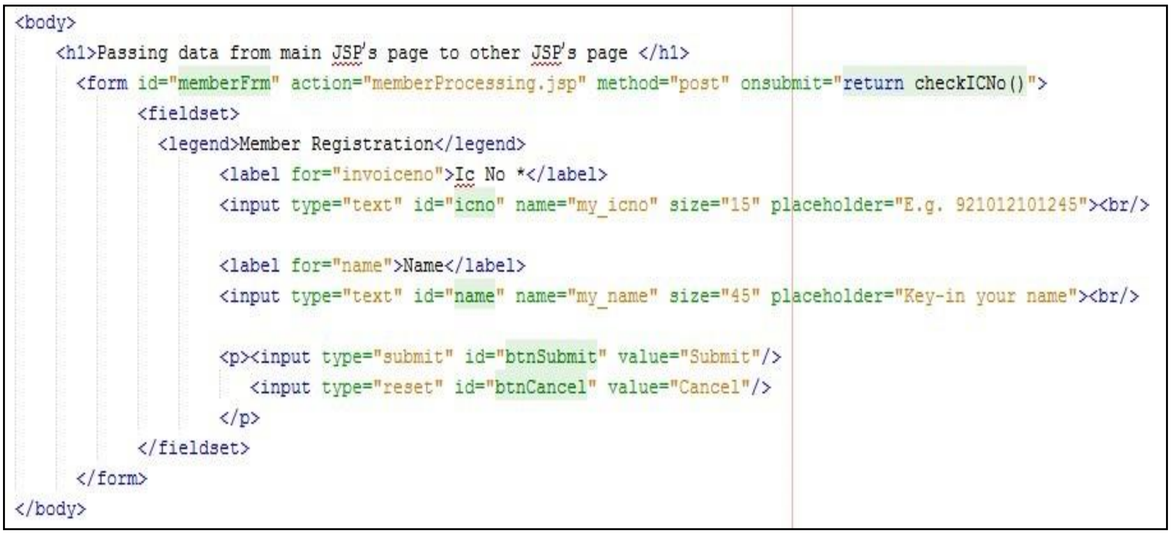
Description:

A JSP file using JSP standard syntax.

4. Click *Finish* button.

5. Source code for *memberRegister.jsp* will appear.

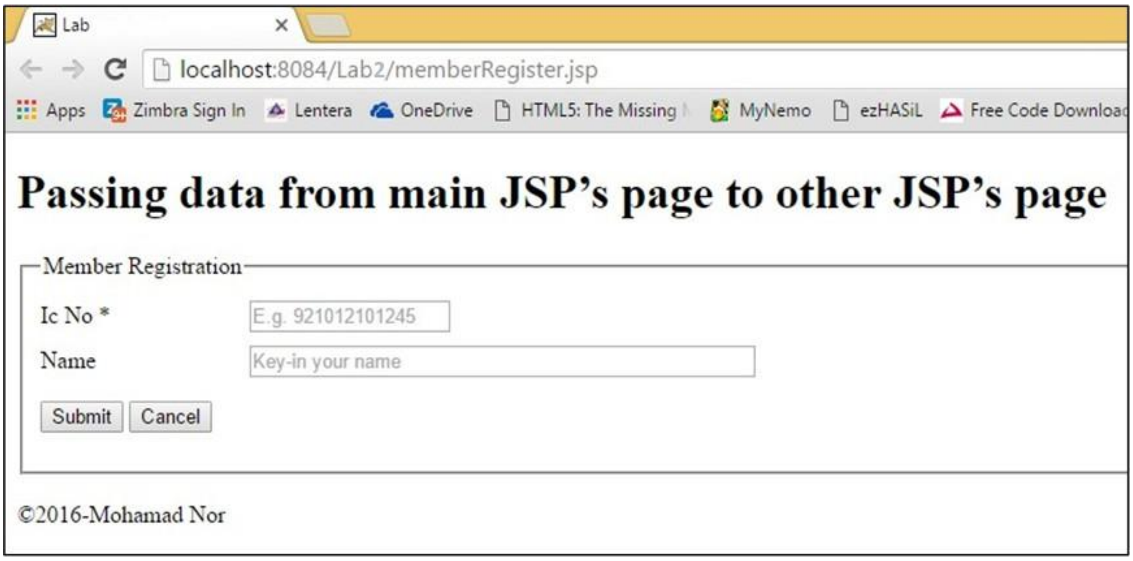
6. Write a HTML's markup to produce HTML's form



```
<body>
    <h1>Passing data from main JSP's page to other JSP's page </h1>
      <form id="memberFrm" action="memberProcessing.jsp" method="post" onsubmit="return checkICNo()">
            <fieldset>
              <legend>Member Registration</legend>
                    <label for="invoiceno">Ic No *</label>
                    <input type="text" id="icno" name="my_icno" size="15" placeholder="E.g. 921012101245"><br/>

                    <label for="name">Name</label>
                    <input type="text" id="name" name="my_name" size="45" placeholder="Key-in your name"><br/>

                    <p><input type="submit" id="btnSubmit" value="Submit"/>
                       <input type="reset" id="btnCancel" value="Cancel"/>
                    </p>
            </fieldset>
      </form>
</body>
```

7. Save and compile *memberRegister.jsp* file.

8. Run the *memberRegister.jsp* file and you should get the interface as below:



Passing data from main JSP's page to other JSP's page

Member Registration

Ic No * E.g. 921012101245

Name Key-in your name

Submit Cancel

©2016-Mohamad Nor

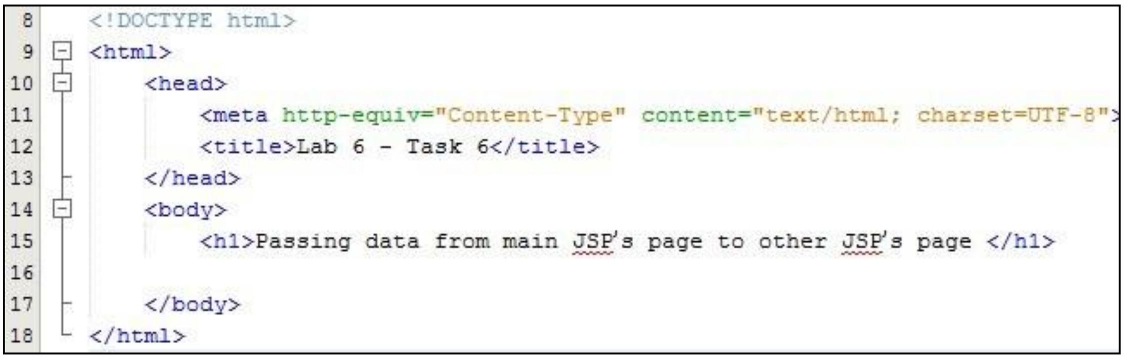
9. Repeat step 1 and step 2.

10. Key-in File Name: *memberProcessing*.

11. Click *Finish* button.

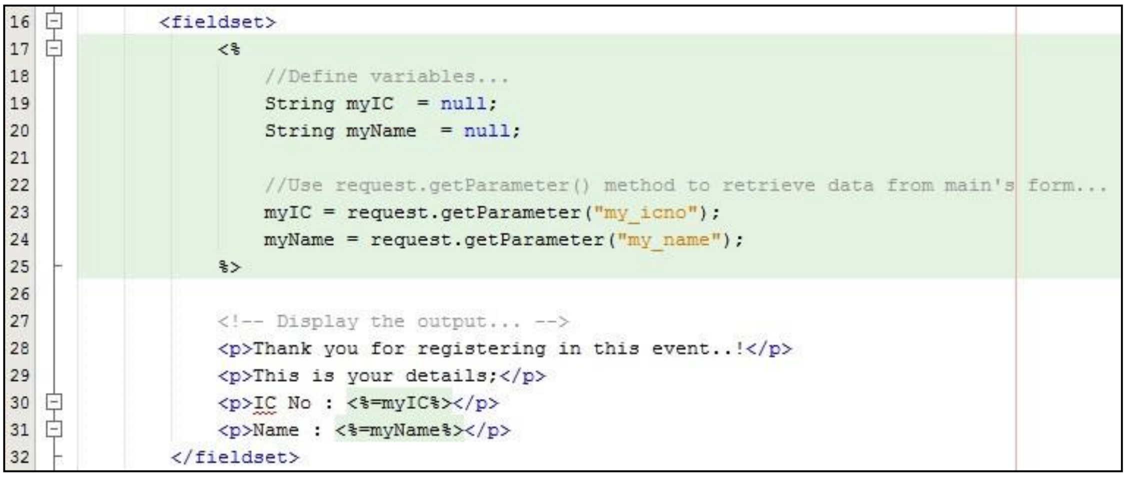
12. Source code for *memberProcessing.jsp* will appear.

13. Write a HTML code.



```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Lab 6 - Task 6</title>
    </head>
    <body>
        <h1>Passing data from main JSP's page to other JSP's page </h1>

    </body>
</html>
```

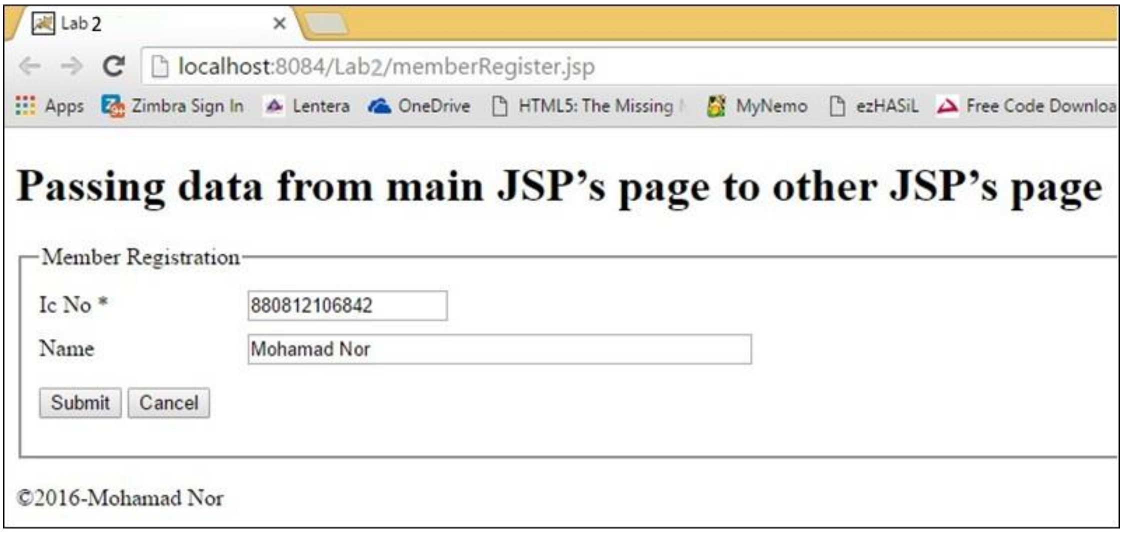
14. Add additional HTML’s tag and Java Scriplet to retrieve the value from main’s form.



```
        <fieldset>
             <%
                 //Define variables...
                 String myIC   = null;
                 String myName  = null;

                 //Use request.getParameter() method to retrieve data from main's form...
                 myIC = request.getParameter("my_icno");
                 myName = request.getParameter("my_name");
             %>

             <!-- Display the output... -->
             <p>Thank you for registering in this event..!</p>
             <p>This is your details;</p>
             <p>IC No : <%=myIC%></p>
             <p>Name : <%=myName%></p>
         </fieldset>
```

15. Compile *memberProcessing.jsp* file.

16. Run the *memberRegister.jsp* file and fill-up the input.



Passing data from main JSP’s page to other JSP’s page

Member Registration

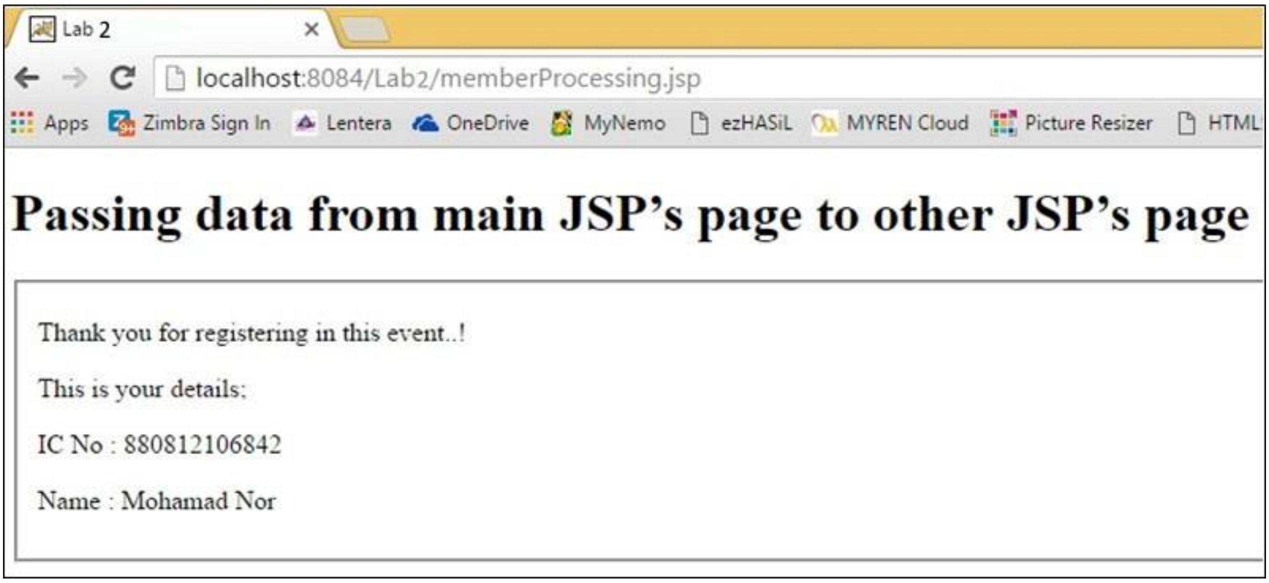
Ic No * 880812106842

Name Mohamad Nor

©2016-Mohamad Nor

17. Click *Submit* button to send the request.

18. These inputs will be sent to *memberProcessing.jsp* page and produce the following page.



Passing data from main JSP's page to other JSP's page

Thank you for registering in this event..!

This is your details:

IC No : 880812106842

Name : Mohamad Nor

Reflection

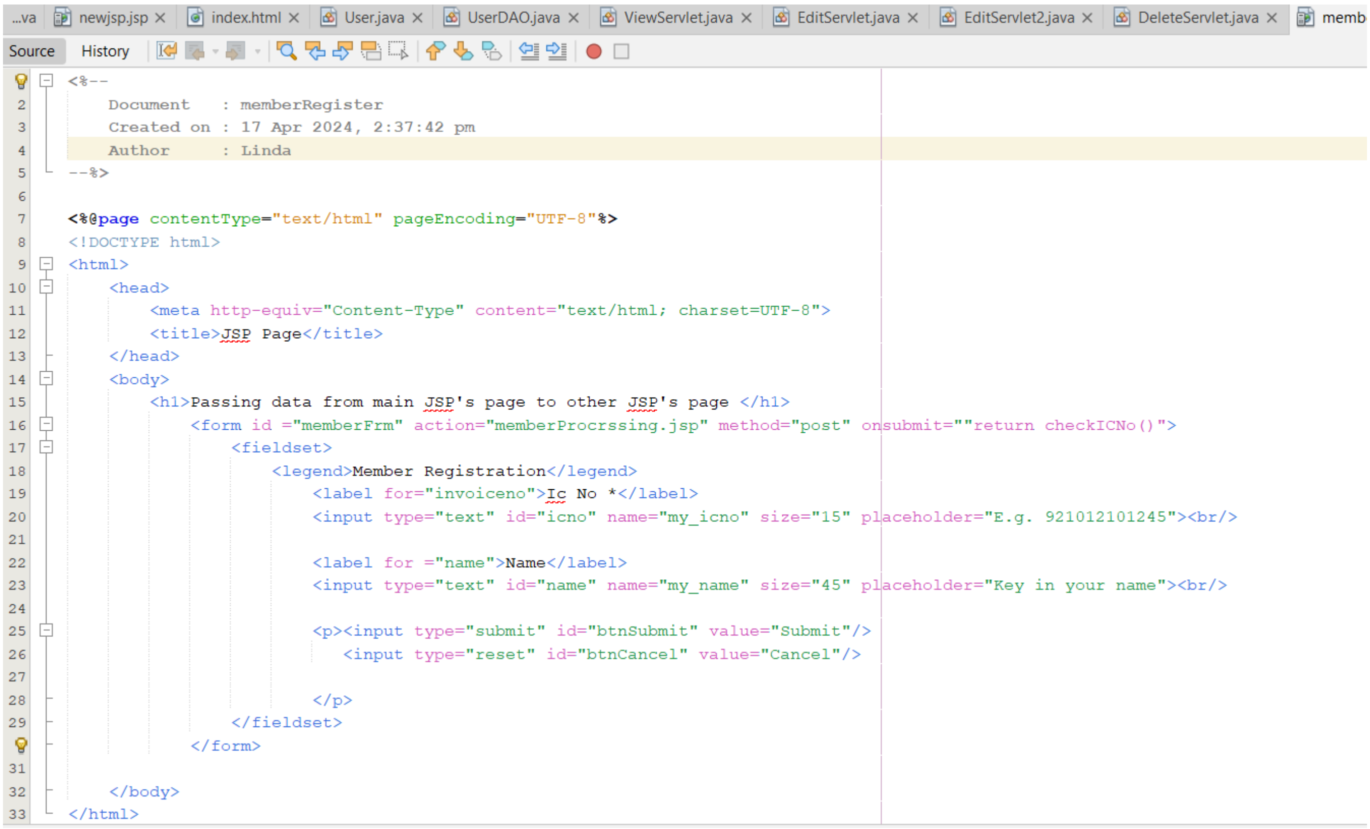
1. How do you want to submit specific information from one form to next form?

use the `action` attribute of the `<form>` tag to specify the URL of the next JSP page. When the form is submitted, the data entered by the user is sent to the URL specified in the `action` attribute using either the GET or POST method.

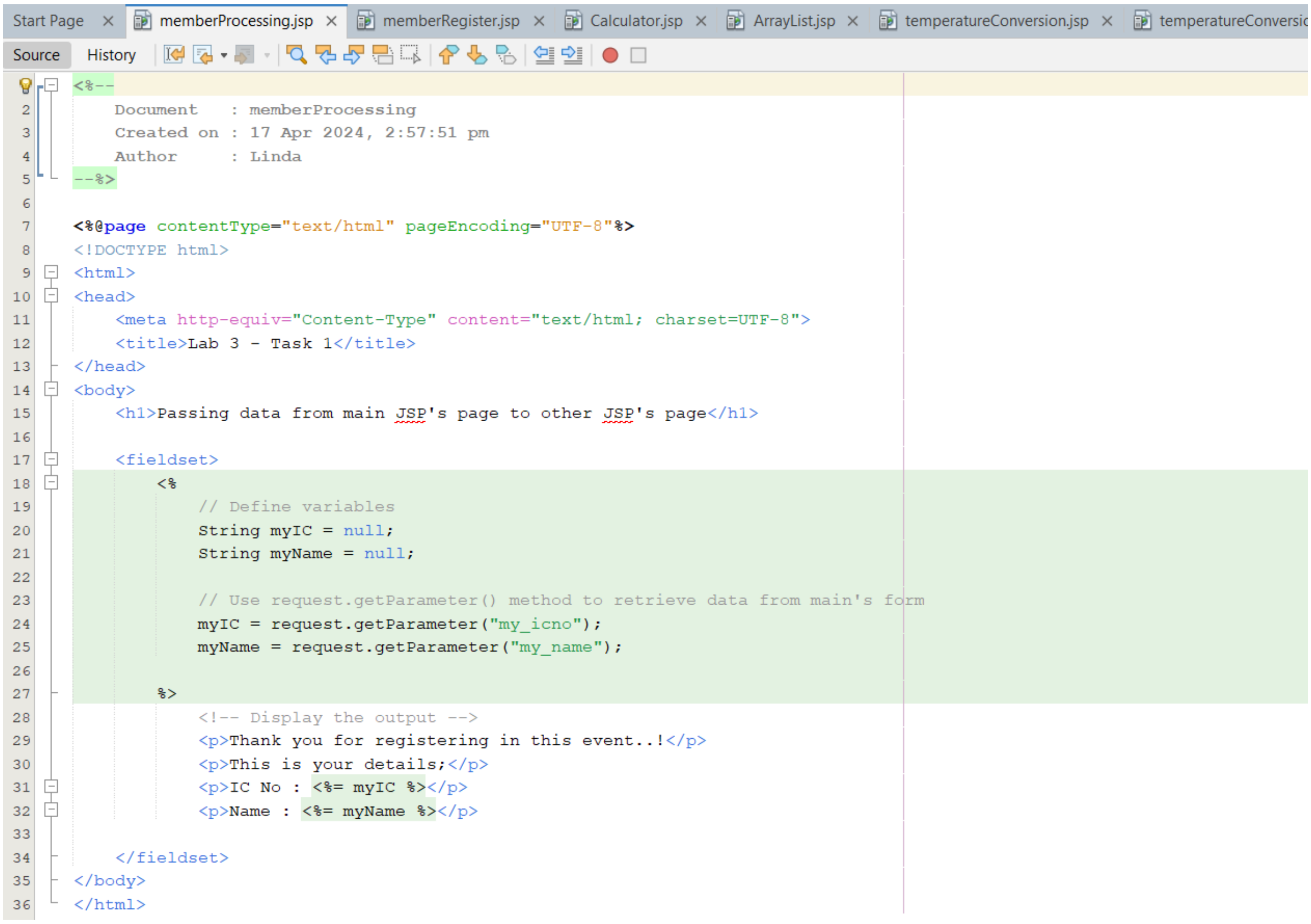
2. What happened if the field name you specify in *request.getParameter("field_name")* in second page is different from the field name you defined in first page?

the value returned by `request.getParameter("field_name")` will be null. This is because `request.getParameter("field_name")` retrieves the value of a request parameter with the given name (`"field_name"` in this case). If there is no parameter with that name or the parameter is not sent in the request, it will return null.

THE CODE:

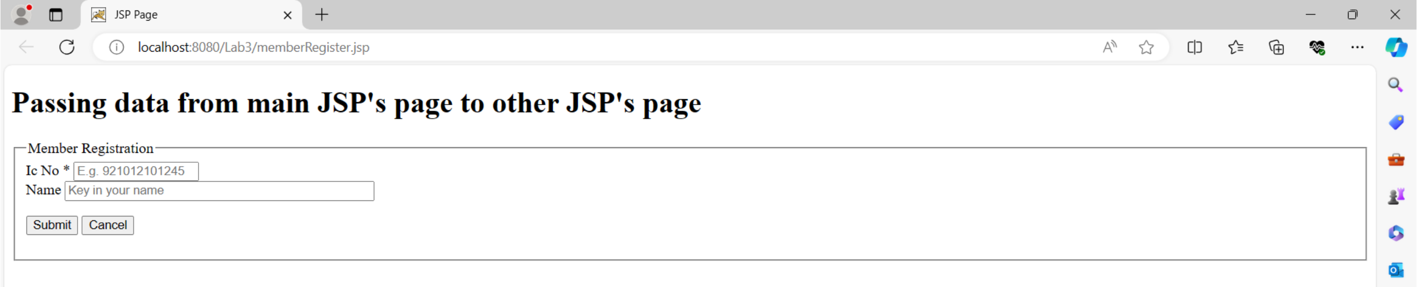


```
<%--
    Document   : memberRegister
    Created on : 17 Apr 2024, 2:37:42 pm
    Author     : Linda
--%>

<%@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>
        <h1>Passing data from main JSP's page to other JSP's page </h1>
            <form id ="memberFrm" action="memberProcrssing.jsp" method="post" onsubmit=""return checkICNo()">
                <fieldset>
                    <legend>Member Registration</legend>
                        <label for="invoiceno">Ic No *</label>
                        <input type="text" id="icno" name="my_icno" size="15" placeholder="E.g. 921012101245"><br/>

                        <label for ="name">Name</label>
                        <input type="text" id="name" name="my_name" size="45" placeholder="Key in your name"><br/>

                        <p><input type="submit" id="btnSubmit" value="Submit"/>
                           <input type="reset" id="btnCancel" value="Cancel"/>

                        </p>
                </fieldset>
            </form>

    </body>
</html>
```



```
<%--
    Document   : memberProcessing
    Created on : 17 Apr 2024, 2:57:51 pm
    Author     : Linda
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Lab 3 - Task 1</title>
</head>
<body>
    <h1>Passing data from main JSP's page to other JSP's page</h1>

    <fieldset>
        <%
            // Define variables
            String myIC = null;
            String myName = null;

            // Use request.getParameter() method to retrieve data from main's form
            myIC = request.getParameter("my_icno");
            myName = request.getParameter("my_name");

        %>
            <!-- Display the output -->
            <p>Thank you for registering in this event..!</p>
            <p>This is your details;</p>
            <p>IC No : <%= myIC %></p>
            <p>Name : <%= myName %></p>

    </fieldset>
</body>
</html>
```

THE OUTPUT:



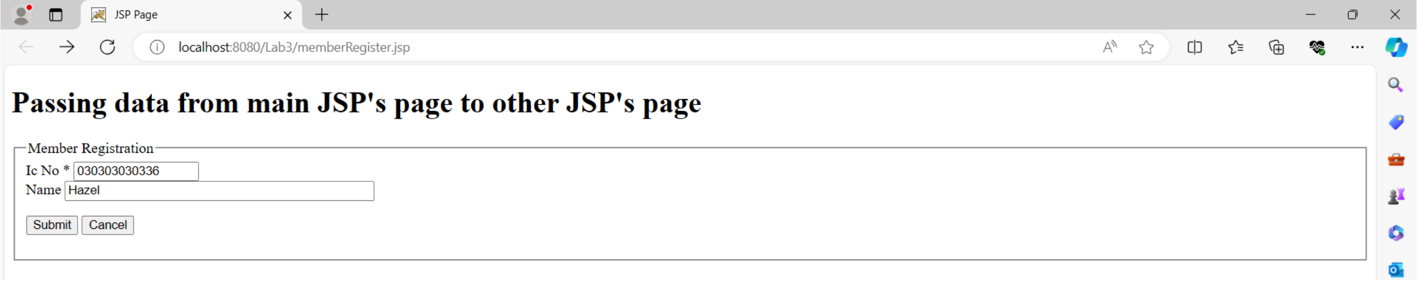
Passing data from main JSP's page to other JSP's page

Member Registration

Ic No * E.g. 921012101245

Name Key in your name

Submit Cancel



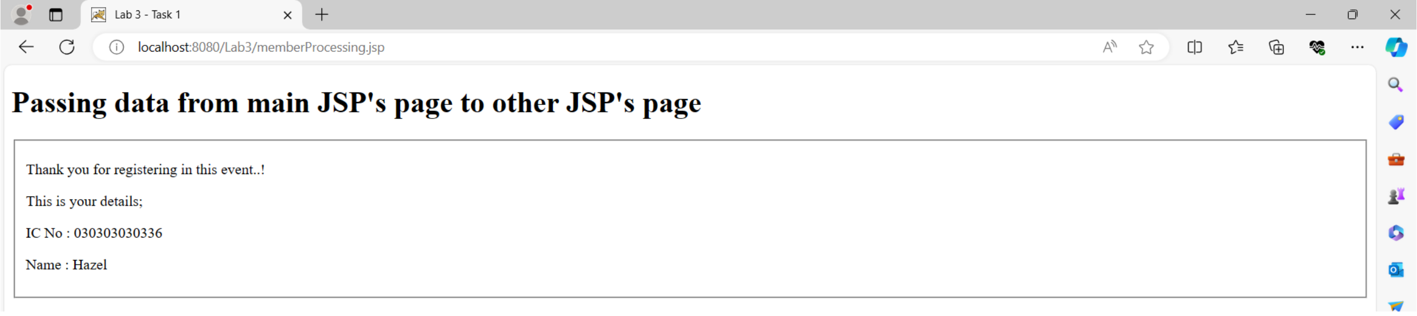
Passing data from main JSP's page to other JSP's page

Member Registration

Ic No * 030303030336

Name Hazel

Submit Cancel



Passing data from main JSP's page to other JSP's page

Thank you for registering in this event..!

This is your details;

IC No : 030303030336

Name : Hazel

Task 2: Using Mathematics Operations in JSP

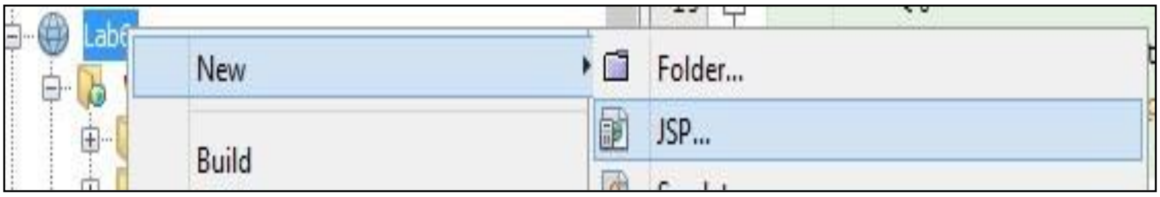
Objective : To demonstrate the use of *request.getParameter (“Mathematics operations”)* in JSP’s page.

Problem Description : i. Create a page *Calculator.jsp* consists of interface represent basic calculator.

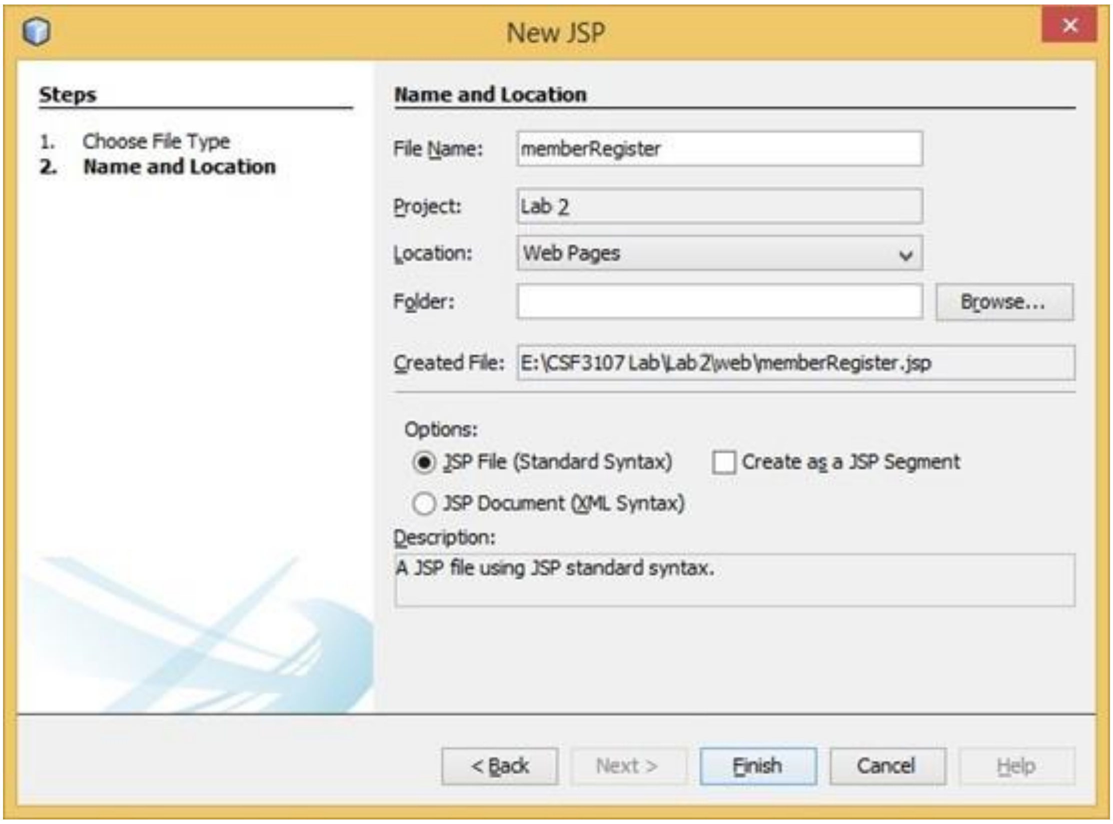
ii. When user key-in inputs, process the request and display the results direct in JSP page.

Estimated time : 30 minutes

1. Go to Project *Lab2*.
2. To create a JSP’s page, right click *Lab2* -> *New* -> *JSP*.



3. Key-in File Name: *Calculator*.



New JSP

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name: memberRegister

Project: Lab 2

Location: Web Pages

Folder:

Created File: E:\CSF3107 Lab\Lab 2\web\memberRegister.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

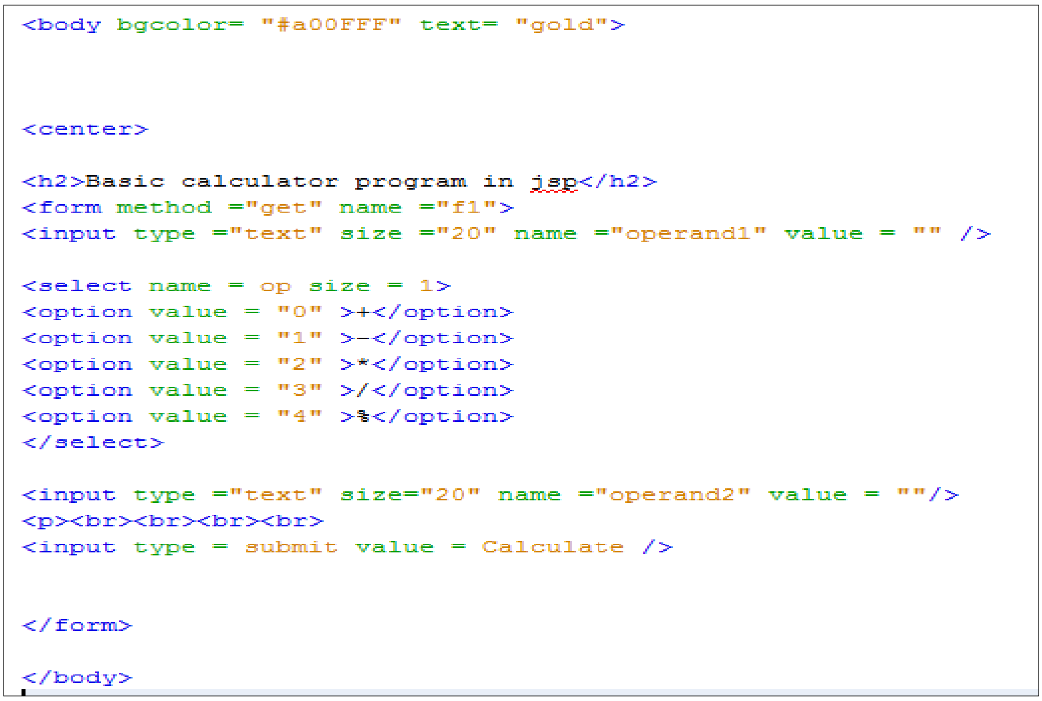
Description:

A JSP file using JSP standard syntax.

4. Click *Finish* button.

5. Source code for *Calculator.jsp* will appear.

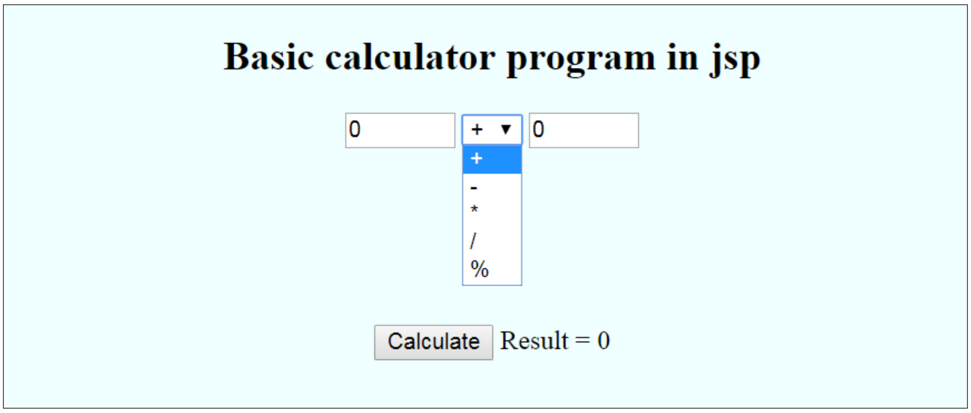
6. Write a HTML's markup to produce HTML's form



```
<body bgcolor= "#a00FFF" text= "gold">



<center>

<h2>Basic calculator program in jsp</h2>
<form method ="get" name ="f1">
<input type ="text" size ="20" name ="operand1" value = "" />

<select name = op size = 1>
<option value = "0" >+</option>
<option value = "1" >-</option>
<option value = "2" >*</option>
<option value = "3" >/</option>
<option value = "4" >%</option>
</select>

<input type ="text" size="20" name ="operand2" value = ""/>
<p><br><br><br><br>
<input type = submit value = Calculate />


</form>

</body>
```

7. Save and compile *Calculator.jsp* file.

8. Run the *Calculator.jsp* file and you should get the interface as below:

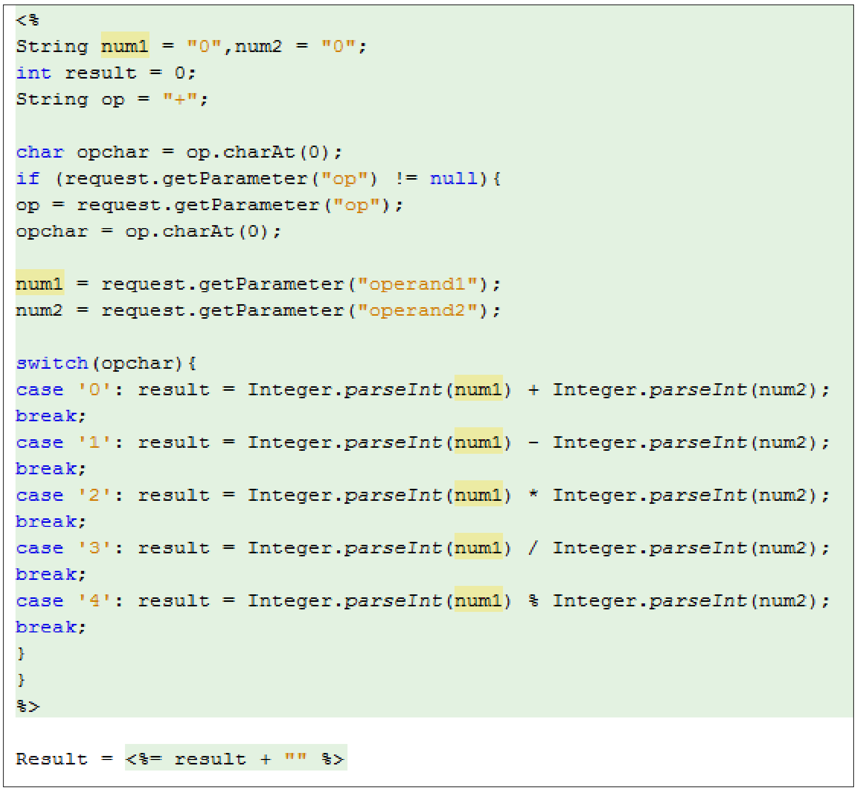


Basic calculator program in jsp

0 + 0

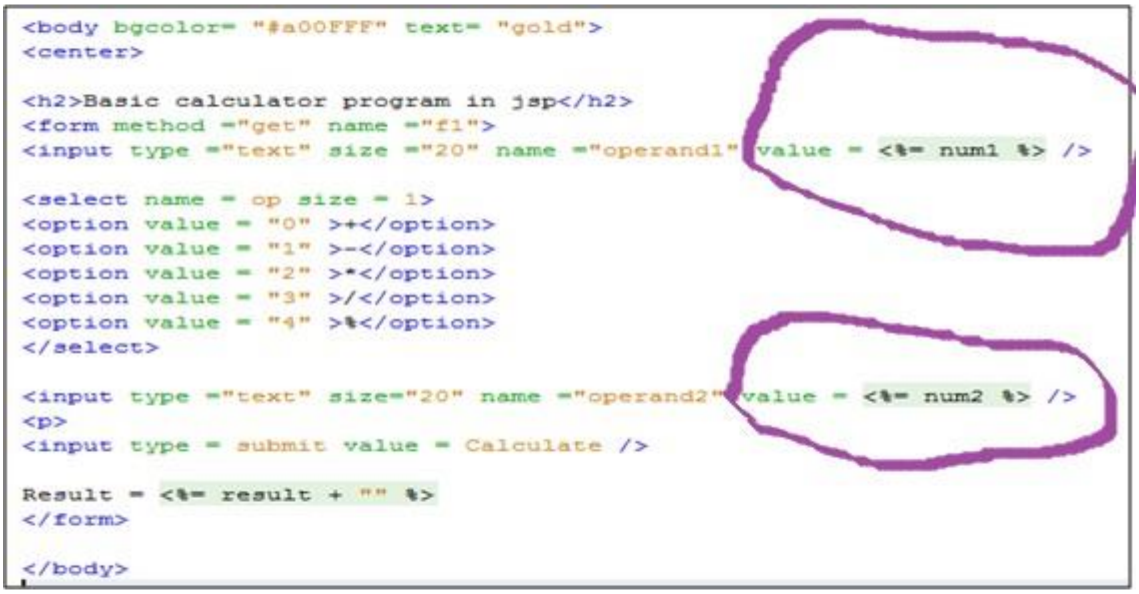
Result = 0

9. Add additional HTML's tag and Java scriplet to retrieve the value from users.



```
<%
String num1 = "0",num2 = "0";
int result = 0;
String op = "+";

char opchar = op.charAt(0);
if (request.getParameter("op") != null){
op = request.getParameter("op");
opchar = op.charAt(0);

num1 = request.getParameter("operand1");
num2 = request.getParameter("operand2");

switch(opchar){
case '0': result = Integer.parseInt(num1) + Integer.parseInt(num2);
break;
case '1': result = Integer.parseInt(num1) - Integer.parseInt(num2);
break;
case '2': result = Integer.parseInt(num1) * Integer.parseInt(num2);
break;
case '3': result = Integer.parseInt(num1) / Integer.parseInt(num2);
break;
case '4': result = Integer.parseInt(num1) % Integer.parseInt(num2);
break;
}
}
%>

Result = <%= result + "" %>
```

9. Further, add additional Java Scriplet to HTML’s tag as below.



```
<body bgcolor= "#a00FFF" text= "gold">
<center>

<h2>Basic calculator program in jsp</h2>
<form method ="get" name ="f1">
<input type ="text" size ="20" name ="operand1" value = <%= num1 %> />

<select name = op size = 1>
<option value = "0" >+</option>
<option value = "1" >-</option>
<option value = "2" >*</option>
<option value = "3" >/</option>
<option value = "4" >%</option>
</select>

<input type ="text" size="20" name ="operand2" value = <%= num2 %> />
<p>
<input type = submit value = Calculate />

Result = <%= result + "" %>
</form>

</body>
```

10. Compile *Calculator.jsp* file.

11. Run the *Calculator.jsp* file and test the clculator.

Reflection

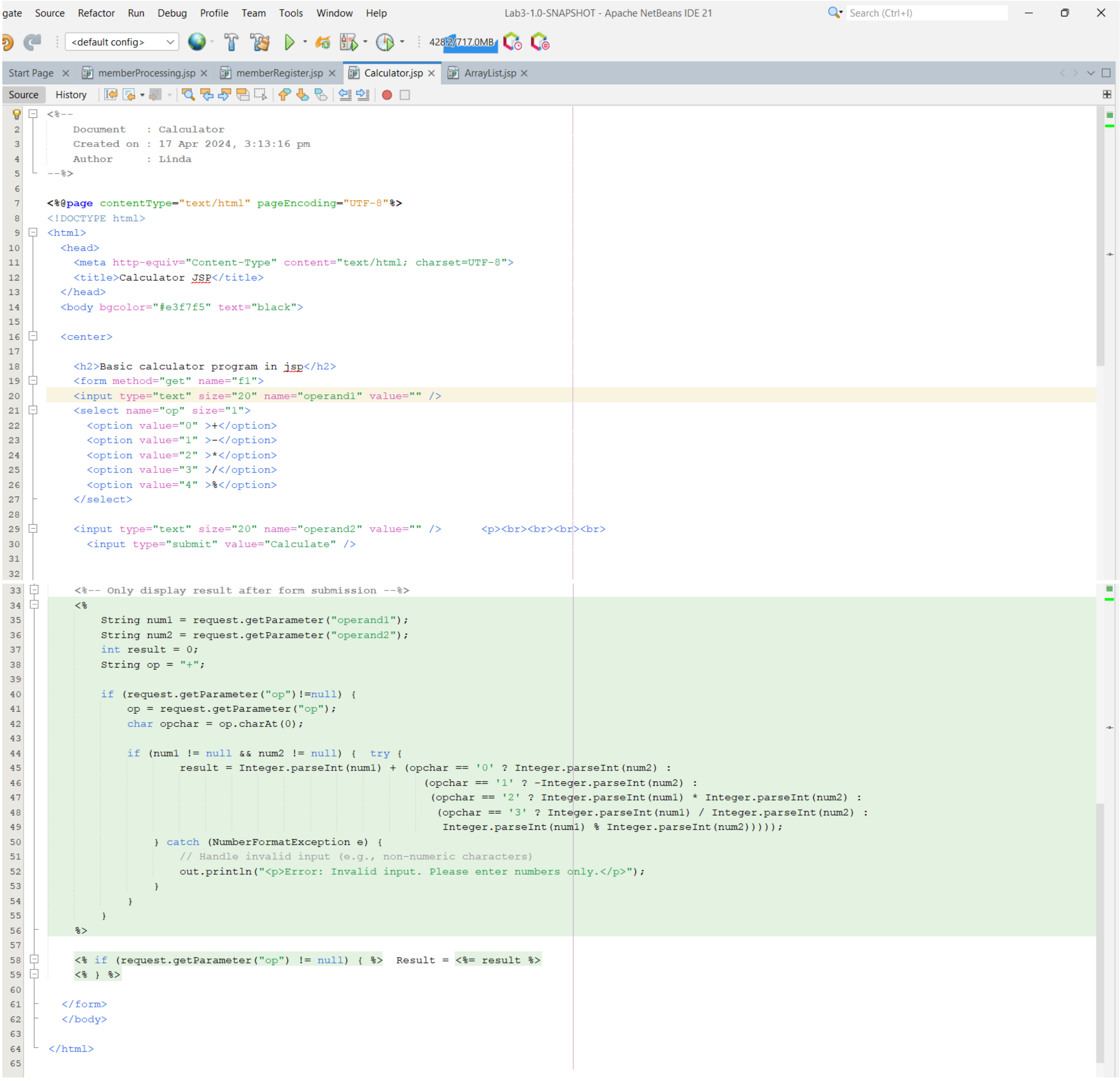
1. How do you want to submit specific information from one form to next form?

   To submit specific information from one form to the next form in a JSP application, you typically use the action attribute of the <form> tag to specify the URL of the next JSP page. When the form is submitted, the data entered by the user is sent to the URL specified in the action attribute using either the GET or POST method.

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

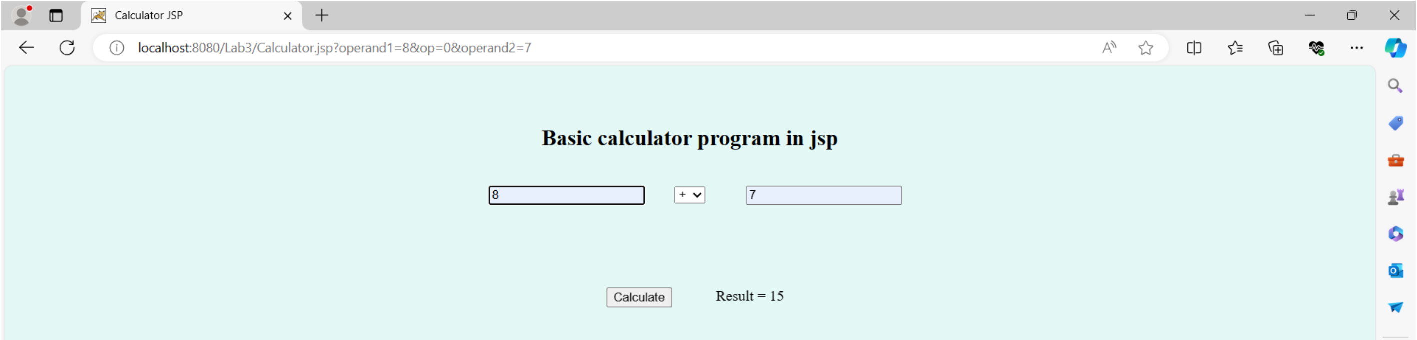
   the value returned by `request.getParameter("field_name")` will be null. This is because `request.getParameter("field_name")` retrieves the value of a request parameter with the given name (`"field_name"` in this case). If there is no parameter with that name or the parameter is not sent in the request, it will return null.

THE CODE:



```
<%--
    Document   : Calculator
    Created on : 17 Apr 2024, 3:13:16 pm
    Author     : Linda
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Calculator JSP</title>
  </head>
  <body bgcolor="#e3f7f5" text="black">

  <center>

    <h2>Basic calculator program in jsp</h2>
    <form method="get" name="f1">
    <input type="text" size="20" name="operand1" value="" />
    <select name="op" size="1">
      <option value="0" >+</option>
      <option value="1" >-</option>
      <option value="2" >*</option>
      <option value="3" >/</option>
      <option value="4" >%</option>
    </select>

    <input type="text" size="20" name="operand2" value="" />      <p><br><br><br><br>
      <input type="submit" value="Calculate" />


    <%-- Only display result after form submission --%>
    <%
        String num1 = request.getParameter("operand1");
        String num2 = request.getParameter("operand2");
        int result = 0;
        String op = "+";

        if (request.getParameter("op")!=null) {
            op = request.getParameter("op");
            char opchar = op.charAt(0);

            if (num1 != null && num2 != null) {  try {
                    result = Integer.parseInt(num1) + (opchar == '0' ? Integer.parseInt(num2) :
                                                      (opchar == '1' ? -Integer.parseInt(num2) :
                                                       (opchar == '2' ? Integer.parseInt(num1) * Integer.parseInt(num2) :
                                                        (opchar == '3' ? Integer.parseInt(num1) / Integer.parseInt(num2) :
                                                         Integer.parseInt(num1) % Integer.parseInt(num2)))));
                } catch (NumberFormatException e) {
                    // Handle invalid input (e.g., non-numeric characters)
                    out.println("<p>Error: Invalid input. Please enter numbers only.</p>");
                }
            }
        }
    %>

    <% if (request.getParameter("op") != null) { %>  Result = <%= result %>
    <% } %>

  </form>
  </body>

</html>

```

THE OUTPUT:


Calculator JSP
localhost:8080/Lab3/Calculator.jsp?operand1=8&op=0&operand2=7

Basic calculator program in jsp

8 + 7

Calculate   Result = 15

Task 4: Perform Calculation of Car Loan

Objective : Passing input to next page for further processing.

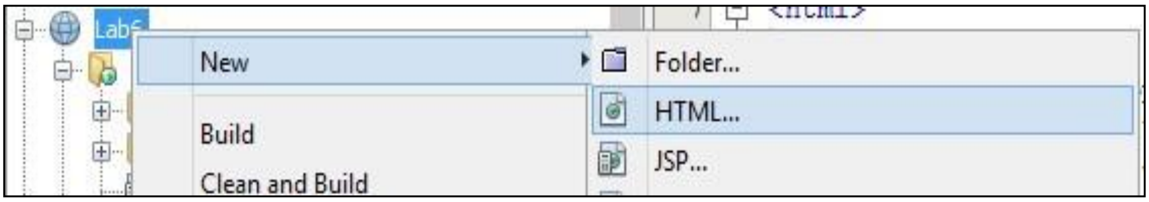
Problem Description : i. Create simple interface in HTML that consists of Loan Amount and Loan Period. (Loan Period < 5 years, interest is 2.8% per year, and > 5 years interest is 4.5% per year).

ii. Submit the form and perform calculation based on user input and, finally, display the result.

Estimated time : 50 minutes

1. Go to Project Lab2.

2. To create a HTML's page, right click Lab2 -> New -> JSP



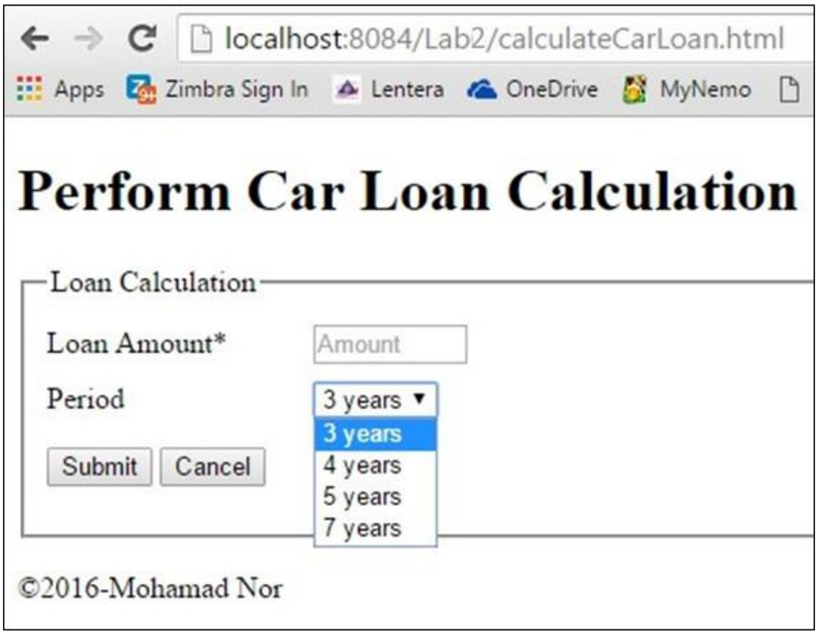
3. Key-in File Name: *calculateCarLoan*

4. Create a standard HTML's markup for form.

5. In your form, create two (2) fields; *Loan Amount* and *Loan Period*.

6. Save *calculateCarLoan.html* and run the file.

7. You will get the following output.



localhost:8084/Lab2/calculateCarLoan.html

Perform Car Loan Calculation

Loan Calculation

Loan Amount* Amount

Period 3 years

©2016-Mohamad Nor

8. Create JSP's file and rename the file as *processCalculateCarLoan*.

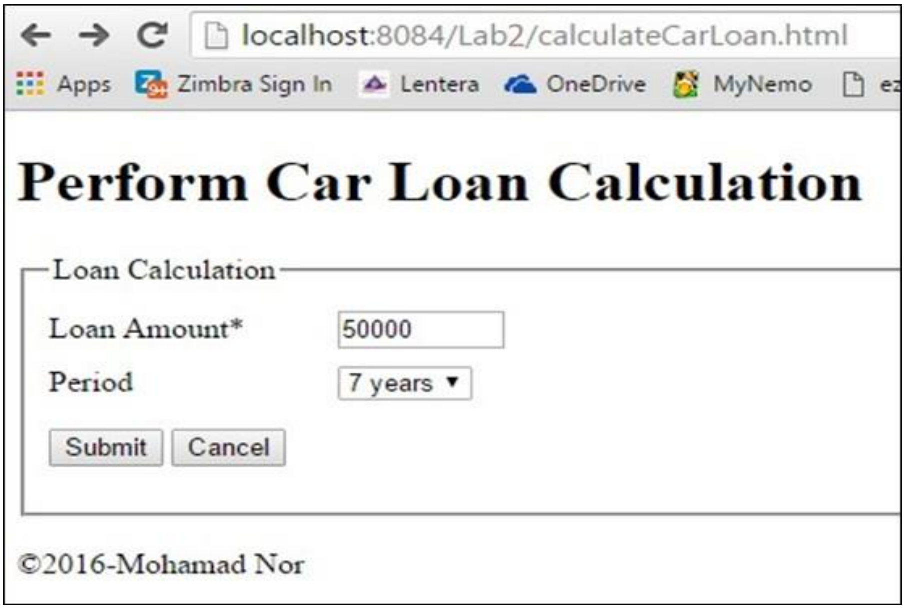
9. Construct the logic for calculating car loan and display the result.

10. Save and compile *processCalculateCarLoan.jsp*.

11. Run *calculateCarLoan.html* file and fill-up the input.

12. Then, submit your result.

13. You should get the following output;



localhost:8084/Lab2/calculateCarLoan.html

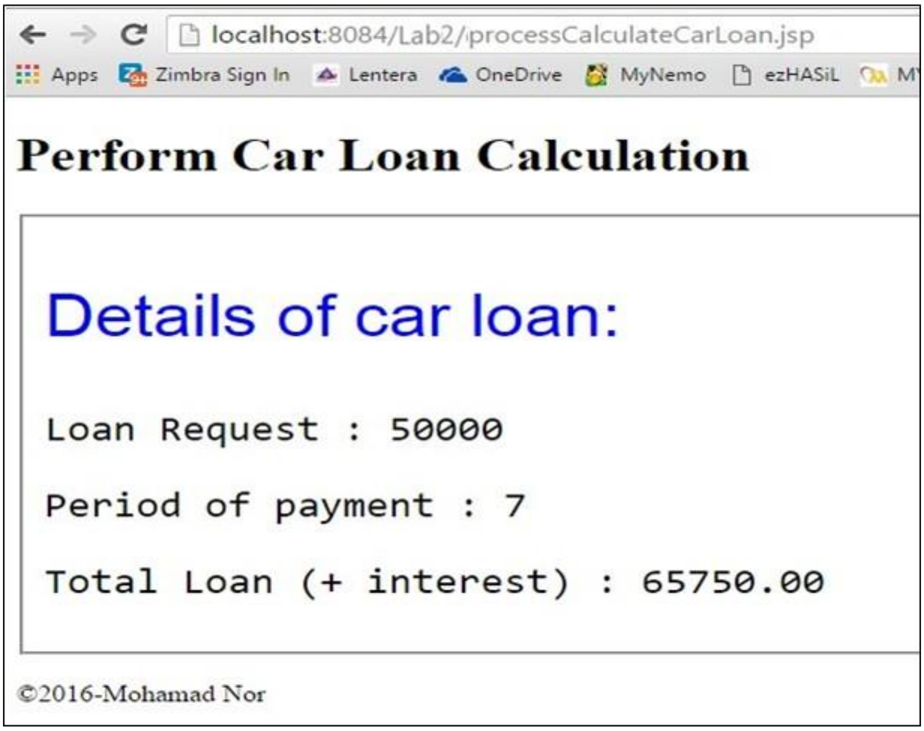
Perform Car Loan Calculation

Loan Calculation

Loan Amount* 50000

Period 7 years

©2016-Mohamad Nor



Perform Car Loan Calculation

Details of car loan:

Loan Request : 50000

Period of payment : 7

Total Loan (+ interest) : 65750.00

©2016-Mohamad Nor

Reflection

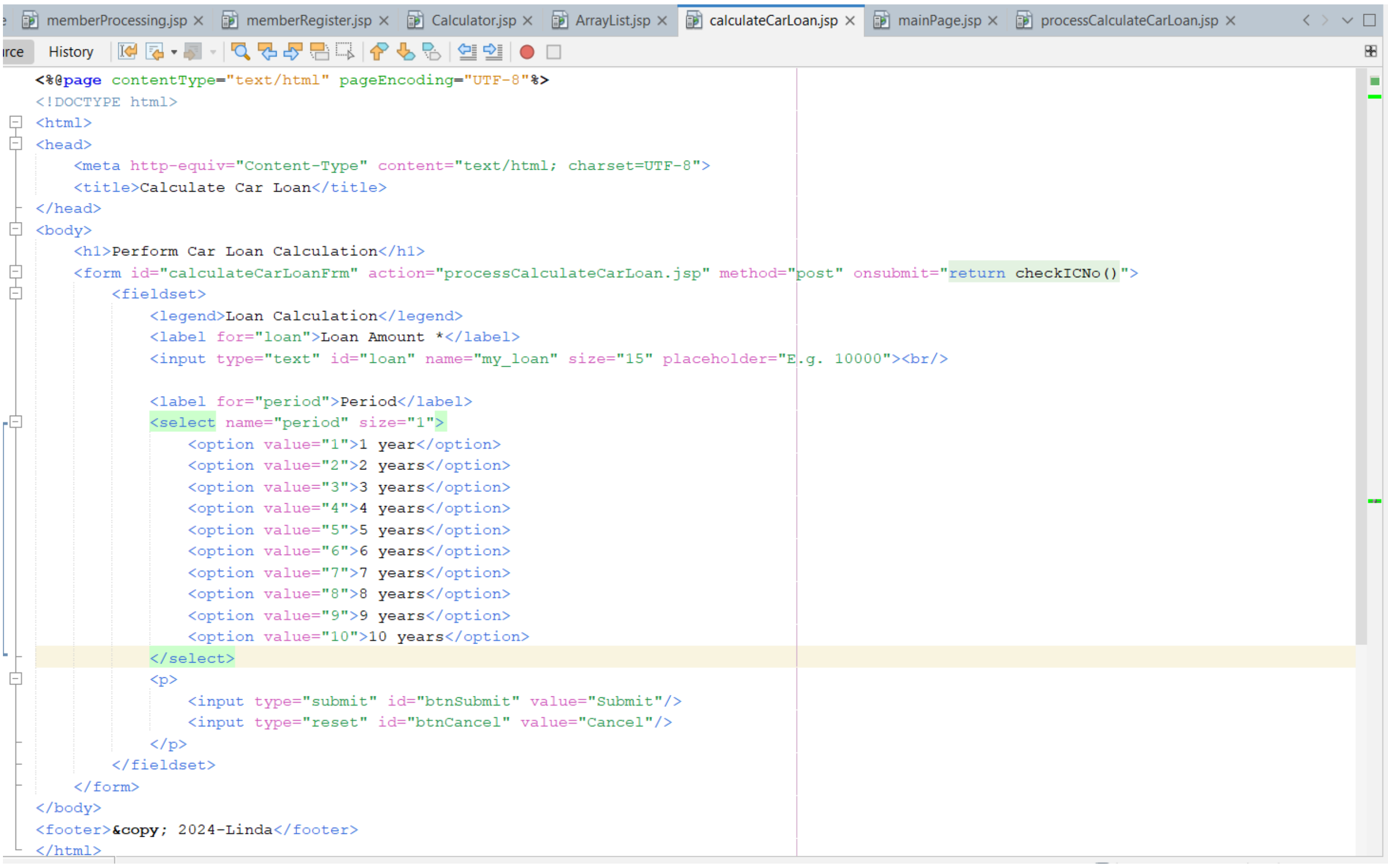
1. How you want to retrieve data from previous page?

   To retrieve data from the previous page in a JSP, you typically use the request.getParameter() method to retrieve the values of form parameters sent via either the GET or POST method. Here's how you can retrieve data from the previous page in your example:In the processCalculateCarLoan.jsp file, you can retrieve the loan amount and period values from the previous page (calculateCarLoan.jsp) using request.getParameter() method

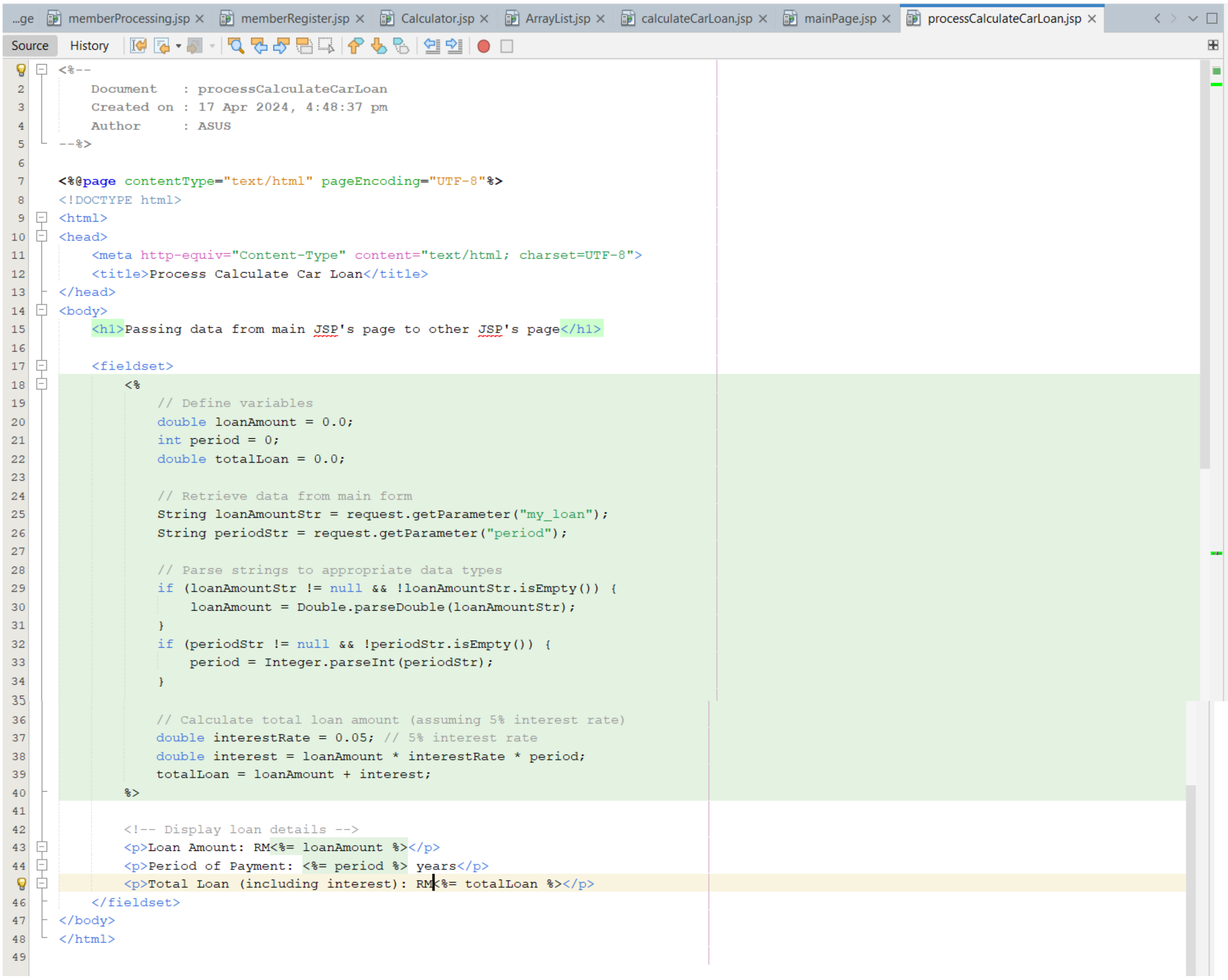
2. Where the construction of logic occur for calculating Total Loan (+ interest) ?

   The construction of logic for calculating the total loan amount (including interest) occurs within the <% %> scriptlet block in the processCalculateCarLoan.jsp file.

THE CODE :

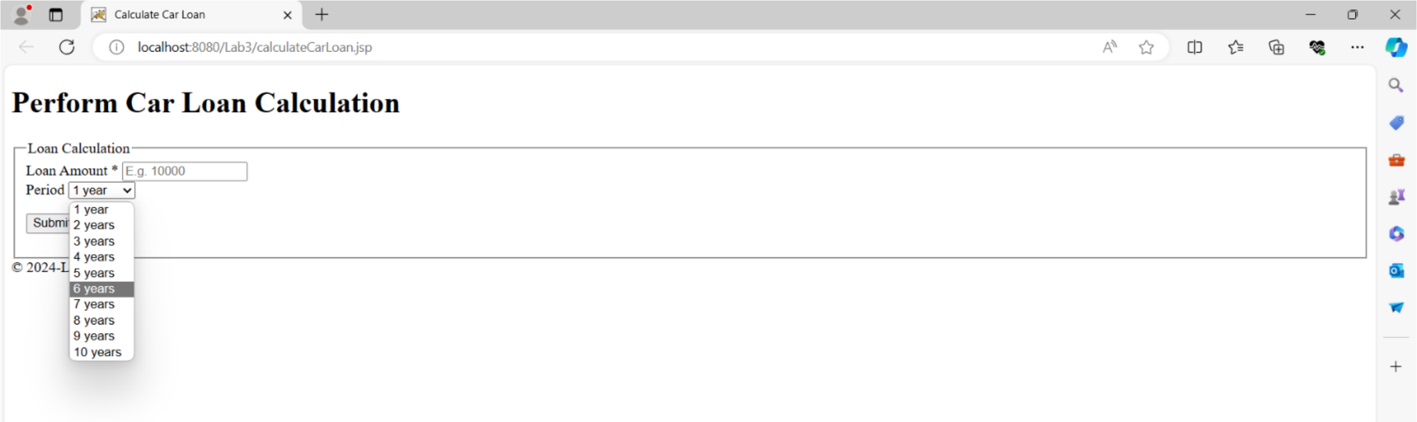


```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Calculate Car Loan</title>
</head>
<body>
    <h1>Perform Car Loan Calculation</h1>
    <form id="calculateCarLoanFrm" action="processCalculateCarLoan.jsp" method="post" onsubmit="return checkICNo()">
        <fieldset>
            <legend>Loan Calculation</legend>
            <label for="loan">Loan Amount *</label>
            <input type="text" id="loan" name="my_loan" size="15" placeholder="E.g. 10000"><br/>

            <label for="period">Period</label>
            <select name="period" size="1">
                <option value="1">1 year</option>
                <option value="2">2 years</option>
                <option value="3">3 years</option>
                <option value="4">4 years</option>
                <option value="5">5 years</option>
                <option value="6">6 years</option>
                <option value="7">7 years</option>
                <option value="8">8 years</option>
                <option value="9">9 years</option>
                <option value="10">10 years</option>
            </select>
            <p>
                <input type="submit" id="btnSubmit" value="Submit"/>
                <input type="reset" id="btnCancel" value="Cancel"/>
            </p>
        </fieldset>
    </form>
</body>
<footer>&copy; 2024-Linda</footer>
</html>
```



```
<%--
    Document   : processCalculateCarLoan
    Created on : 17 Apr 2024, 4:48:37 pm
    Author     : ASUS
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Process Calculate Car Loan</title>
</head>
<body>
    <h1>Passing data from main JSP's page to other JSP's page</h1>

    <fieldset>
        <%
            // Define variables
            double loanAmount = 0.0;
            int period = 0;
            double totalLoan = 0.0;

            // Retrieve data from main form
            String loanAmountStr = request.getParameter("my_loan");
            String periodStr = request.getParameter("period");

            // Parse strings to appropriate data types
            if (loanAmountStr != null && !loanAmountStr.isEmpty()) {
                loanAmount = Double.parseDouble(loanAmountStr);
            }
            if (periodStr != null && !periodStr.isEmpty()) {
                period = Integer.parseInt(periodStr);
            }

            // Calculate total loan amount (assuming 5% interest rate)
            double interestRate = 0.05; // 5% interest rate
            double interest = loanAmount * interestRate * period;
            totalLoan = loanAmount + interest;
        %>

        <!-- Display loan details -->
        <p>Loan Amount: RM<%= loanAmount %></p>
        <p>Period of Payment: <%= period %> years</p>
        <p>Total Loan (including interest): RM<%= totalLoan %></p>
    </fieldset>
</body>
</html>
```

THE OUTPUT:



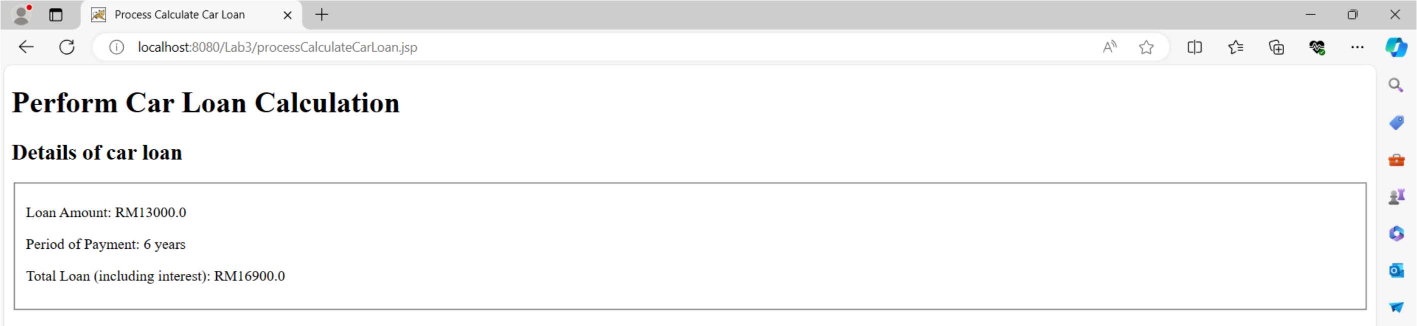
Perform Car Loan Calculation

Loan Calculation

Loan Amount * E.g. 10000

Period 1 year

© 2024-L


Process Calculate Car Loan
localhost:8080/Lab3/processCalculateCarLoan.jsp

Perform Car Loan Calculation

Details of car loan

Loan Amount: RM13000.0

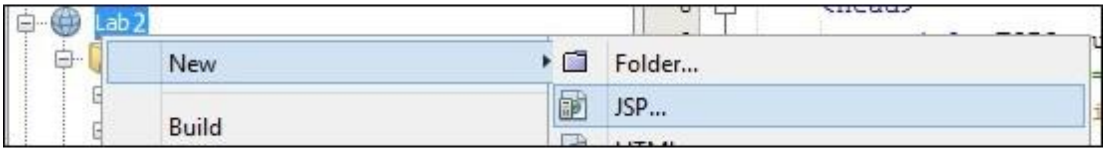
Period of Payment: 6 years

Total Loan (including interest): RM16900.0

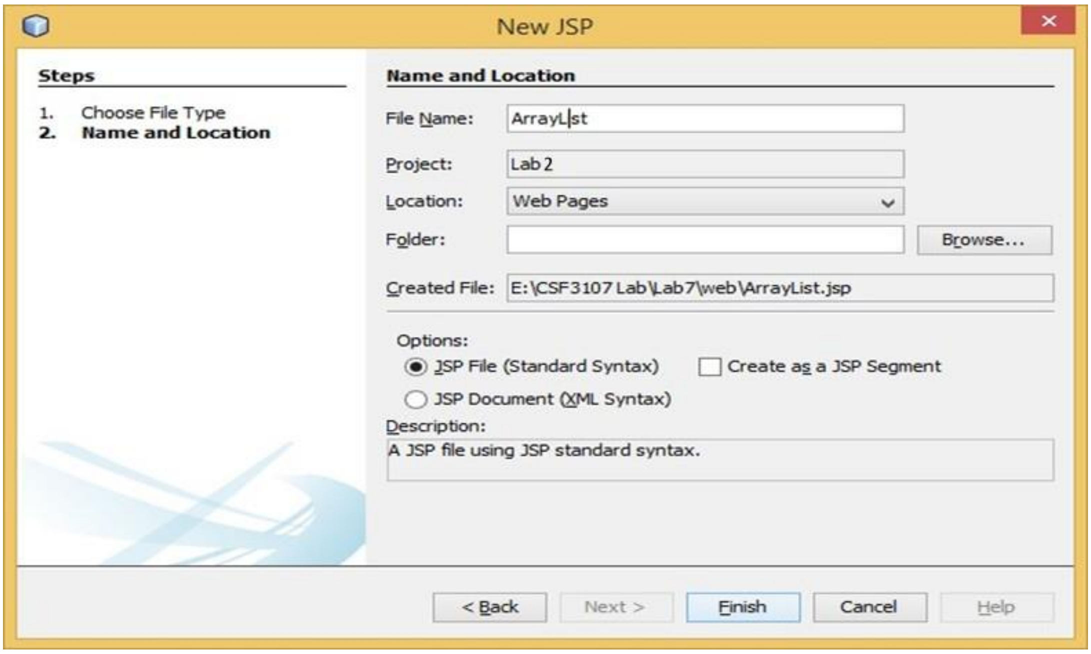
Task 5: Using JSP Page Directive to Call Java API

Objective	:	Use JSP page directive elements to call certain Java API.
Problem Description	:	Using Java *ArrayList* object to store data and retrive it via JSP page.
Estimated time	:	20 minutes

1. Create a new JSP's file.



2. Type file name as *ArrayList*.



File Name: ArrayList

Project: Lab 2

Location: Web Pages

Folder:

Created File: E:\CSF3107 Lab\Lab7\web\ArrayList.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

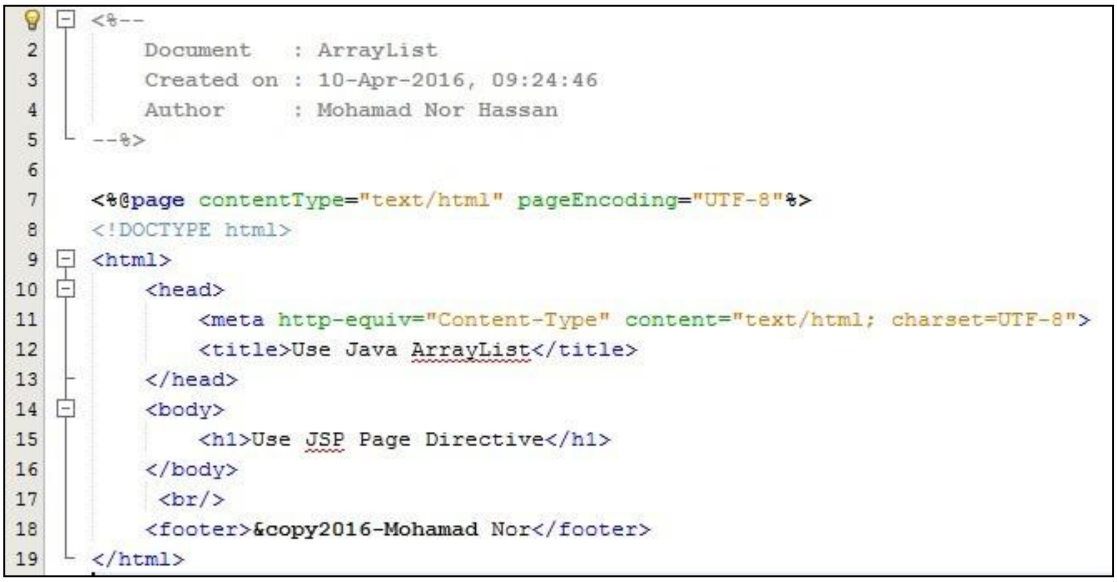
Description:

A JSP file using JSP standard syntax.

2. Click *Finish* button.

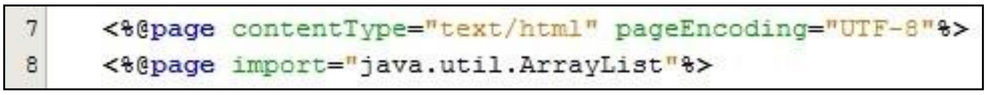
4. Type title as *Use Java ArrayList*.

5. Type header1 as *Use JSP Page Directive*



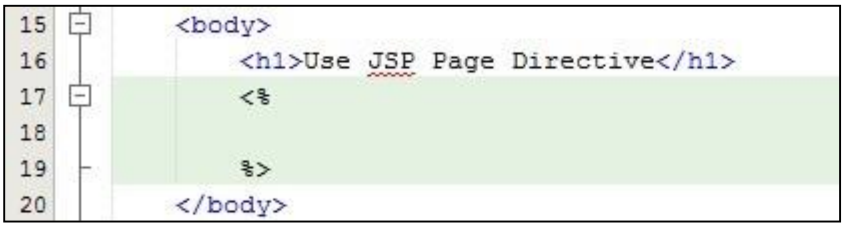
```
<%--
    Document   : ArrayList
    Created on : 10-Apr-2016, 09:24:46
    Author     : Mohamad Nor Hassan
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Use Java ArrayList</title>
    </head>
    <body>
        <h1>Use JSP Page Directive</h1>
    </body>
     <br/>
    <footer>&copy2016-Mohamad Nor</footer>
</html>
```

6. In order to use Java *ArrayList*'s object, we need to use JSP Page Directive and import the related API.



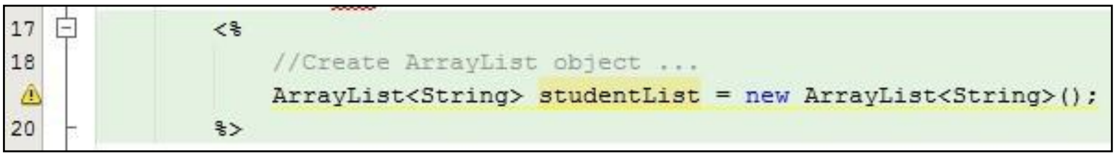
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.ArrayList"%>
```

7. In order to use Java syntax, create a Java Scriptlet notation.



```
    <body>
        <h1>Use JSP Page Directive</h1>
        <%

        %>
    </body>
```

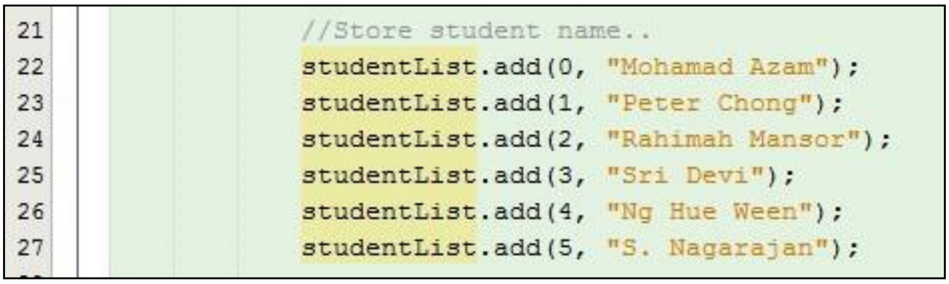
8. Create an object *ArrayList* to store a list of student name.



```
        <%
            //Create ArrayList object ...
            ArrayList<String> studentList = new ArrayList<String>();
        %>
```

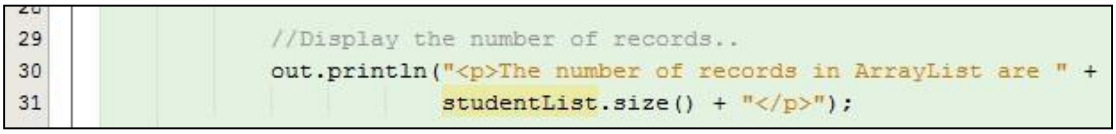
9. Add the following name to ArrayList's object.

- ✓ Mohamad Azam
- ✓ Peter Chong
- ✓ Rahimah Mansor
- ✓ Sri Devi
- ✓ Ng Hue Ween
- ✓ S. Nagarajan



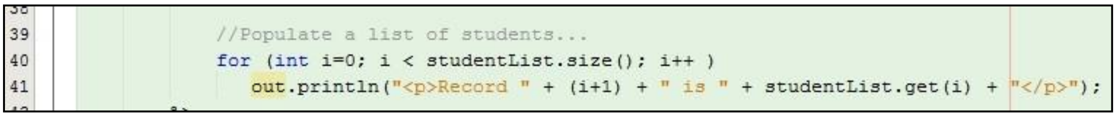
```
//Store student name..
studentList.add(0, "Mohamad Azam");
studentList.add(1, "Peter Chong");
studentList.add(2, "Rahimah Mansor");
studentList.add(3, "Sri Devi");
studentList.add(4, "Ng Hue Ween");
studentList.add(5, "S. Nagarajan");
```

10. Display the number of records for an ArrayList's object.



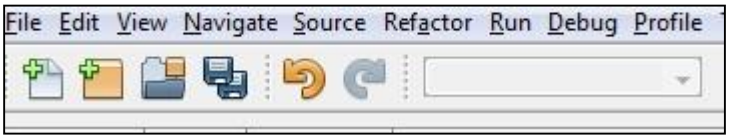
```
//Display the number of records..
out.println("<p>The number of records in ArrayList are " +
            studentList.size() + "</p>");
```

11. Finally, populate the list of students.

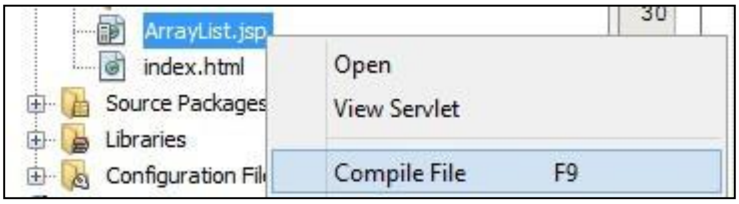


```
//Populate a list of students...
for (int i=0; i < studentList.size(); i++ )
   out.println("<p>Record " + (i+1) + " is " + studentList.get(i) + "</p>");
```

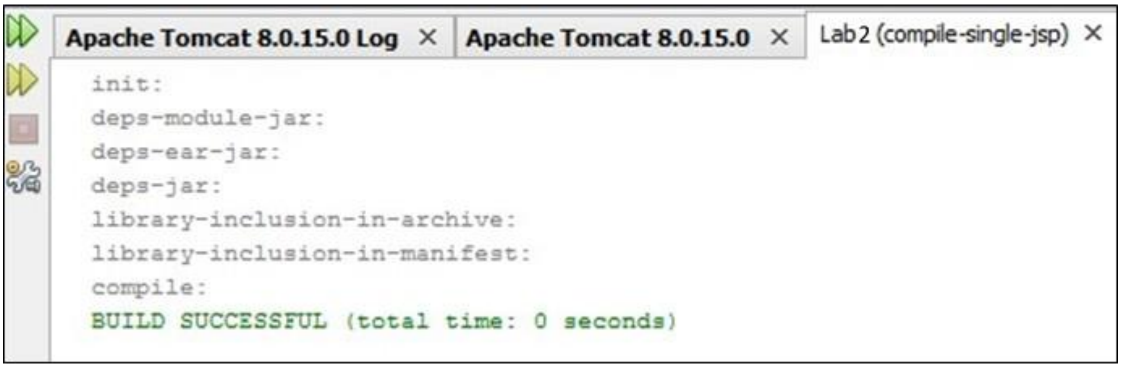
12. Click *SaveAll* icon.



13. Right click file *ArrayList.jsp* and click *Compile File* (F9).



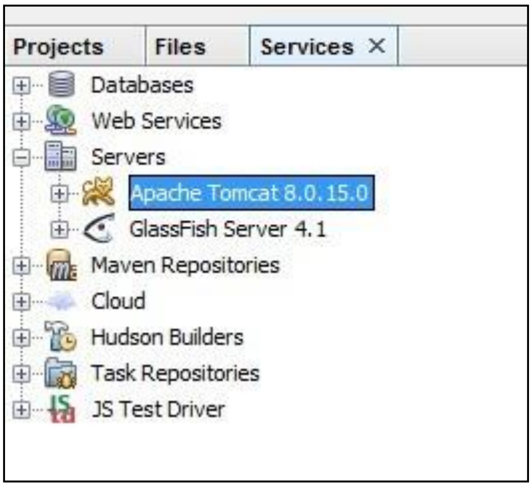
14. You will get notification message the the bottom of Netbeans IDE with the green colour.



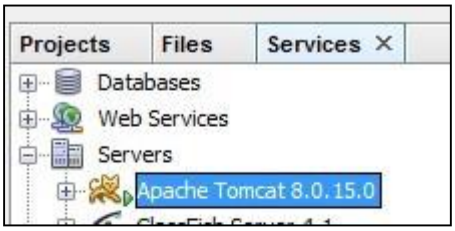
```
init:
deps-module-jar:
deps-ear-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
BUILD SUCCESSFUL (total time: 0 seconds)
```

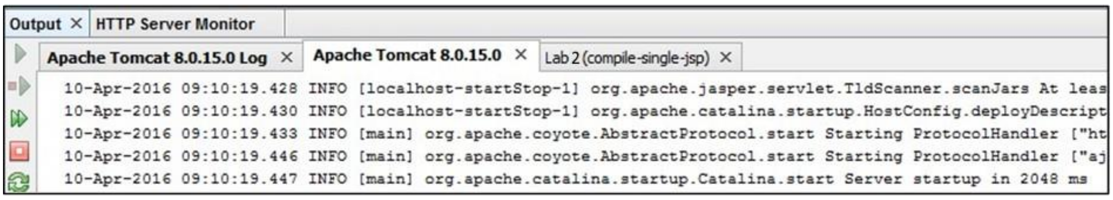
15. Before running any JSP's files for first time upon opening your Netbeans IDE, you need to start your web server (i.e; Apache Tomcat).

16. To perform this, go to *Services* -> *Servers* -> *Apache Tomcat.*



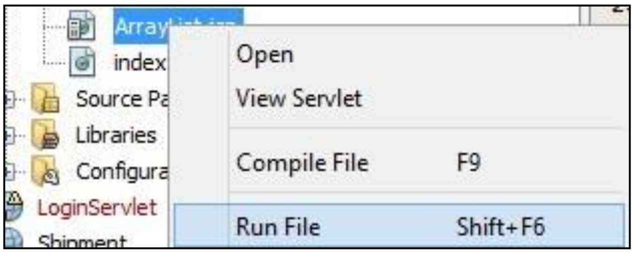
17. You should get the green indicator at *Apache Tomcat*'s icon and *Apache Tomcat* output message with the time taken to start specified time to start *Apache Tomcat* web server.



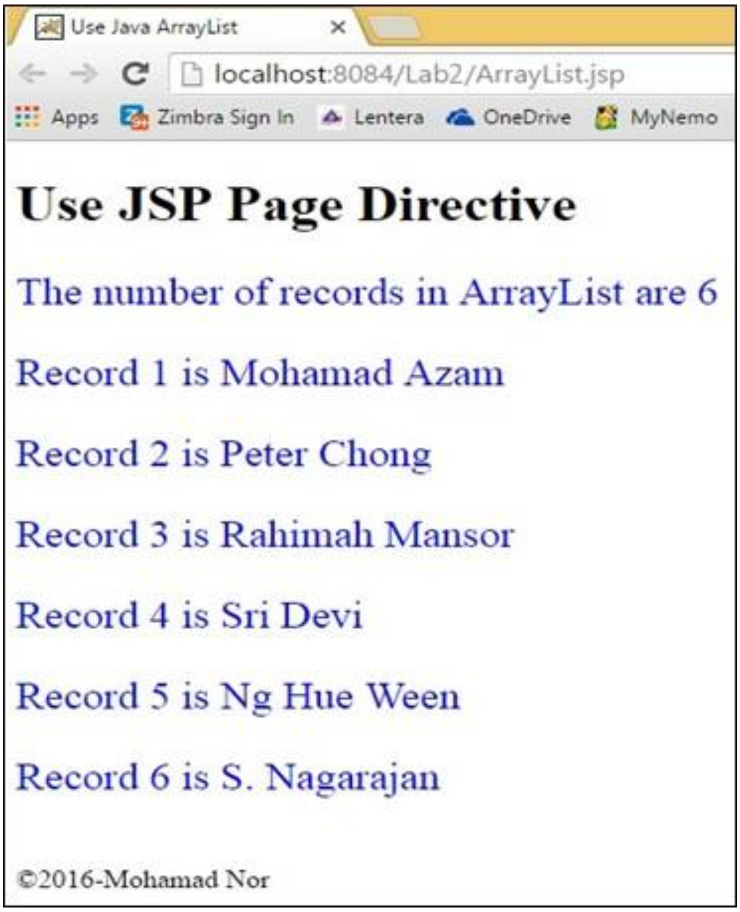


```
10-Apr-2016 09:10:19.428 INFO [localhost-startStop-1] org.apache.jasper.servlet.TldScanner.scanJars At leas
10-Apr-2016 09:10:19.430 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDescript
10-Apr-2016 09:10:19.433 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ht
10-Apr-2016 09:10:19.446 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["aj
10-Apr-2016 09:10:19.447 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 2048 ms
```

18. Go *to Project*'s tab. Then right click file *ArrayList.jsp* and click *Run File* (Shift+F6).



19. Output will appear in web browser.



Use JSP Page Directive

The number of records in ArrayList are 6

Record 1 is Mohamad Azam

Record 2 is Peter Chong

Record 3 is Rahimah Mansor

Record 4 is Sri Devi

Record 5 is Ng Hue Ween

Record 6 is S. Nagarajan

©2016-Mohamad Nor

Reflection

1. What you have learnt from this exercise?

   learn how to utilize JSP page directive elements to incorporate certain Java APIs into your JSP pages. This allows you to leverage Java's extensive library ecosystem directly within your JSP code, enhancing the functionality and capabilities of your web applications.

2. Write a sample syntax how you want to use java *Math* object in JSP?

```
<%@ page import="java.lang.Math" %>

<%
   double number = 5.5;
   double result = Math.sqrt(number); // Example usage of Math.sqrt() method
    %>
```

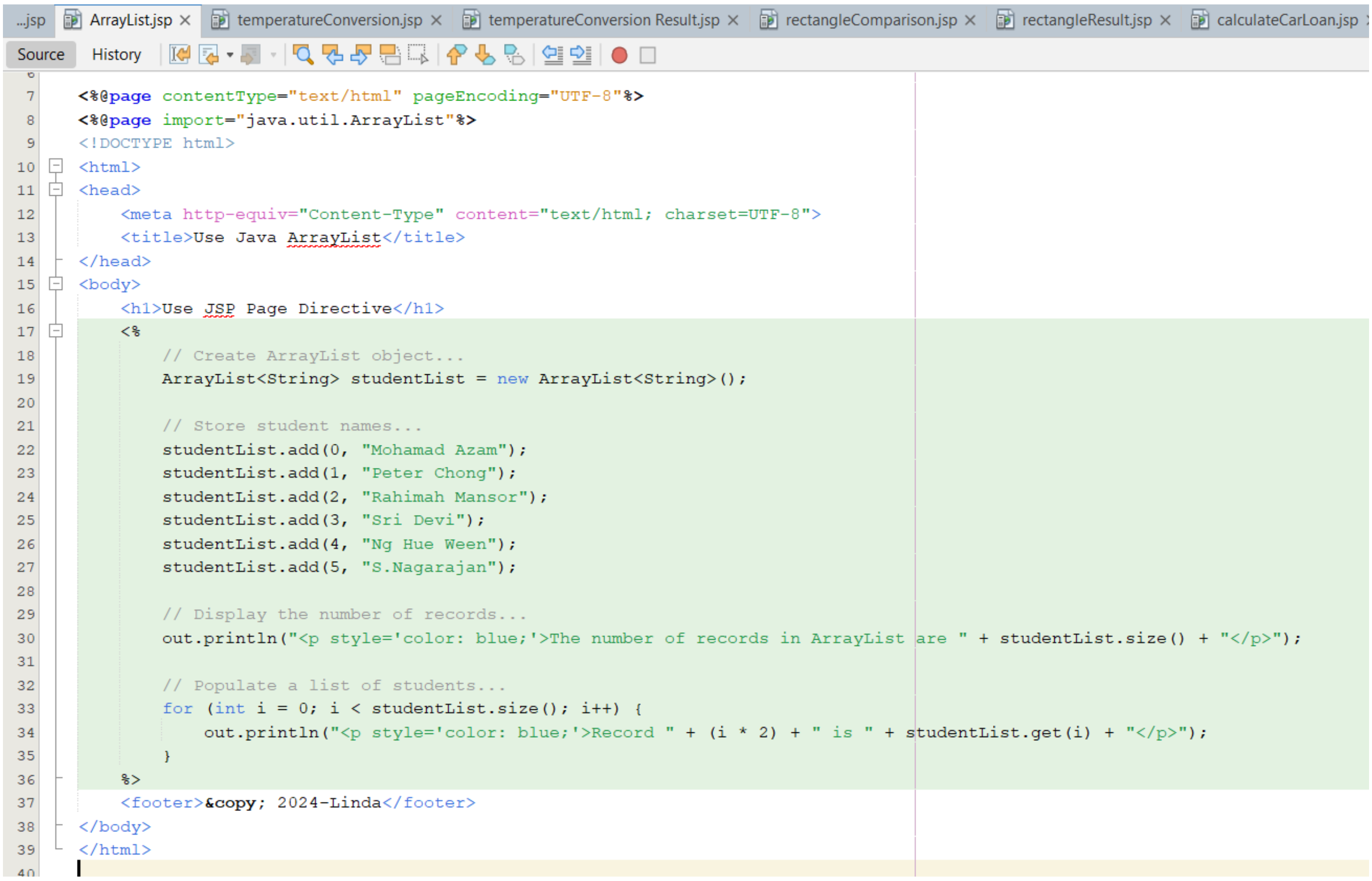
3. List and write a sample syntax for THREE (3) of JSP page directive.

   - Page Import Directive: Used to import Java classes or packages into the JSP file.
     `<%@ page import="java.util.ArrayList" %>`

   - Page ContentType Directive: Specifies the content type and character encoding for the response.
     `<%@ page contentType="text/html; charset=UTF-8" %>`

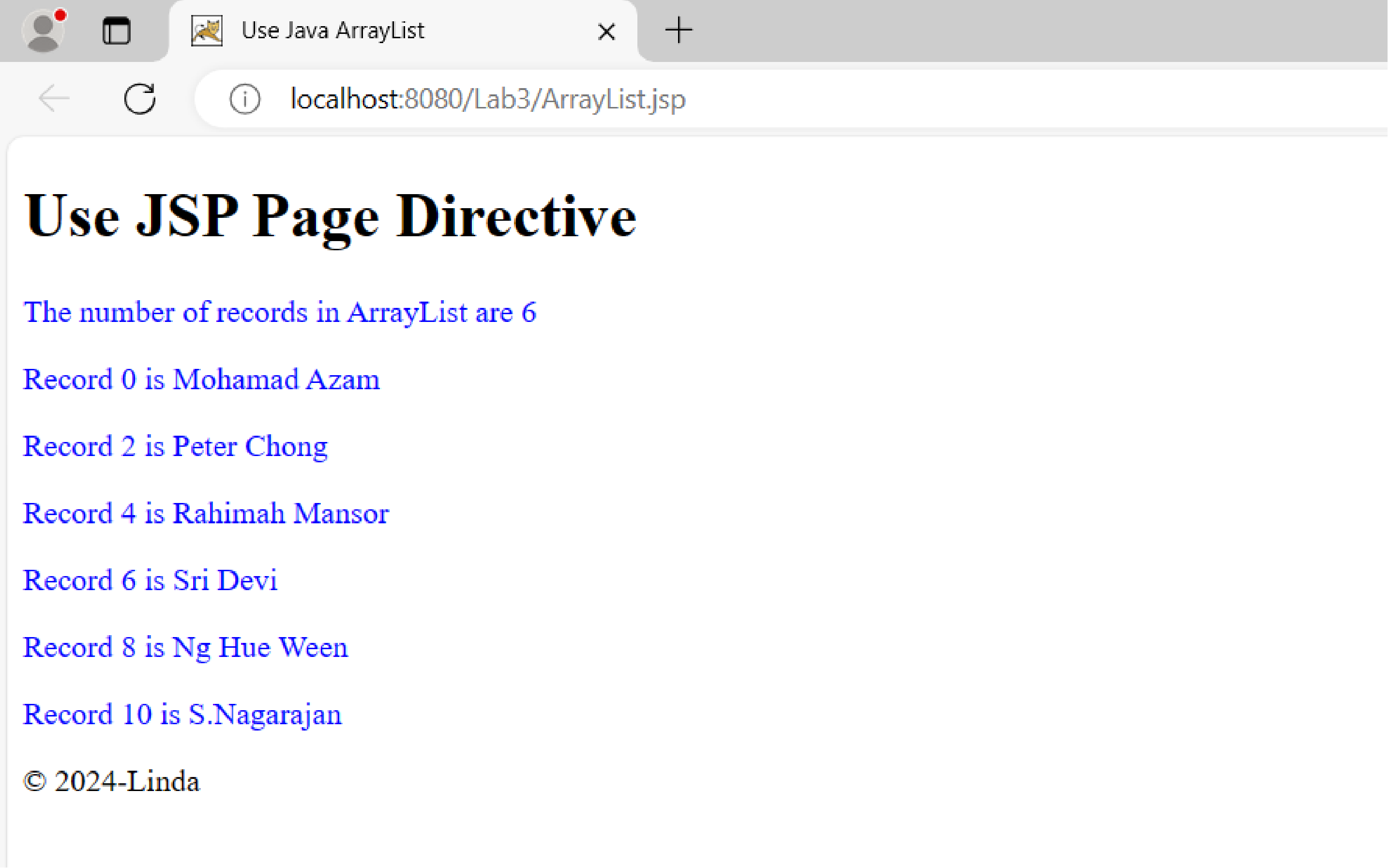
   - Page Encoding Directive: Specifies the character encoding used for the JSP page.
     `<%@ page pageEncoding="UTF-8" %>`

THE CODE:



```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.ArrayList"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Use Java ArrayList</title>
</head>
<body>
    <h1>Use JSP Page Directive</h1>
    <%
        // Create ArrayList object...
        ArrayList<String> studentList = new ArrayList<String>();

        // Store student names...
        studentList.add(0, "Mohamad Azam");
        studentList.add(1, "Peter Chong");
        studentList.add(2, "Rahimah Mansor");
        studentList.add(3, "Sri Devi");
        studentList.add(4, "Ng Hue Ween");
        studentList.add(5, "S.Nagarajan");

        // Display the number of records...
        out.println("<p style='color: blue;'>The number of records in ArrayList are " + studentList.size() + "</p>");

        // Populate a list of students...
        for (int i = 0; i < studentList.size(); i++) {
            out.println("<p style='color: blue;'>Record " + (i * 2) + " is " + studentList.get(i) + "</p>");
        }
    %>
    <footer>&copy; 2024-Linda</footer>
</body>
</html>
```

THE OUTPUT:



Use JSP Page Directive

The number of records in ArrayList are 6

Record 0 is Mohamad Azam

Record 2 is Peter Chong

Record 4 is Rahimah Mansor

Record 6 is Sri Devi

Record 8 is Ng Hue Ween

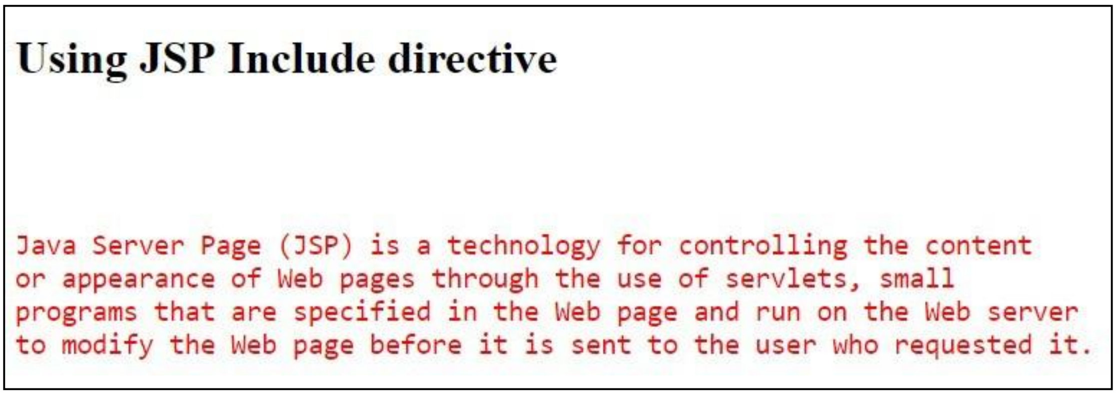
Record 10 is S.Nagarajan

© 2024-Linda

Task 6: Use JSP Include directive for JSP Page

Objective	:	Demonstrate the use of JSP Page Include directive.
Problem	:	Create a JSP master page that displays the header, main contents and footer.
Description		
Estimated time	:	30 minutes

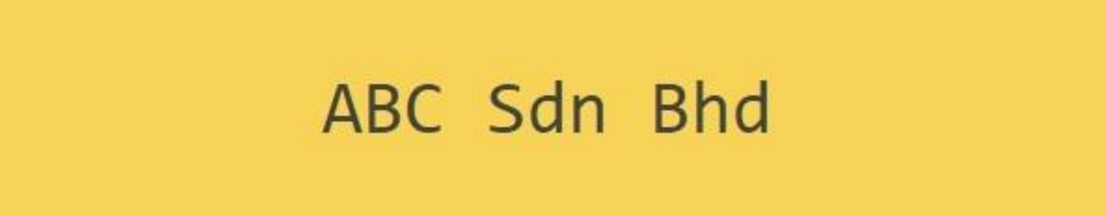
1. Create a new JSP's file.
2. Type file name as *mainPage*.
3. Create content for *mainPage.jsp* as below.



Using JSP Include directive

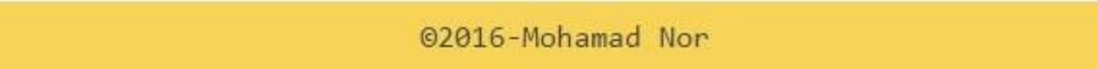
```
Java Server Page (JSP) is a technology for controlling the content
or appearance of Web pages through the use of servlets, small
programs that are specified in the Web page and run on the Web server
to modify the Web page before it is sent to the user who requested it.
```

4. Create a header file as *headerPage.jsp* and display the following output.



ABC Sdn Bhd

5. Create a header file as *footerPage.jsp* and display the following output



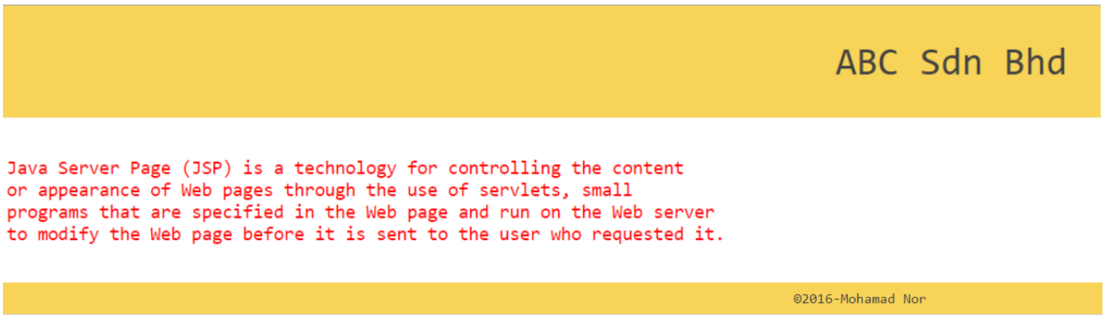
©2016-Mohamad Nor

6. Include your *headerPage.jsp* and *footerPage.jsp* inside your *mainPage.jsp*.

7. Save *mainPage.jsp*

8. Compile and run *mainPage.jsp*.

9. You should het the following output.



ABC Sdn Bhd

```
Java Server Page (JSP) is a technology for controlling the content
or appearance of Web pages through the use of servlets, small
programs that are specified in the Web page and run on the Web server
to modify the Web page before it is sent to the user who requested it.
```

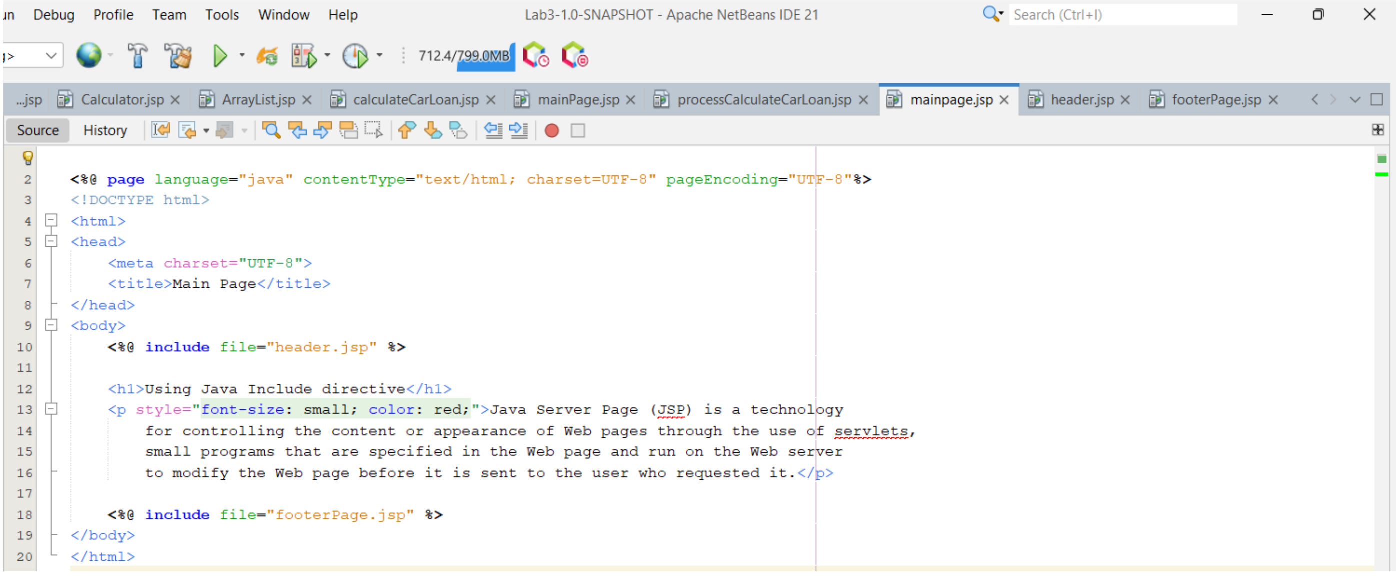
©2016-Mohamad Nor

Reflection

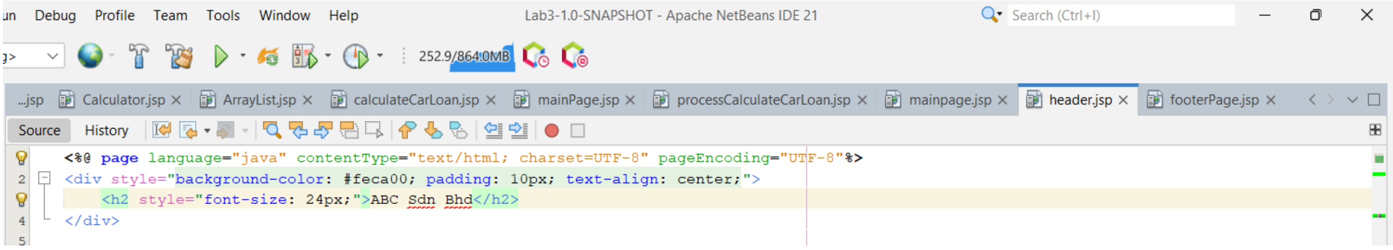
1. What you have learnt from this exercise?

2. Write a syntax how you want to include *common.html* file that located at a directory known as *master*.

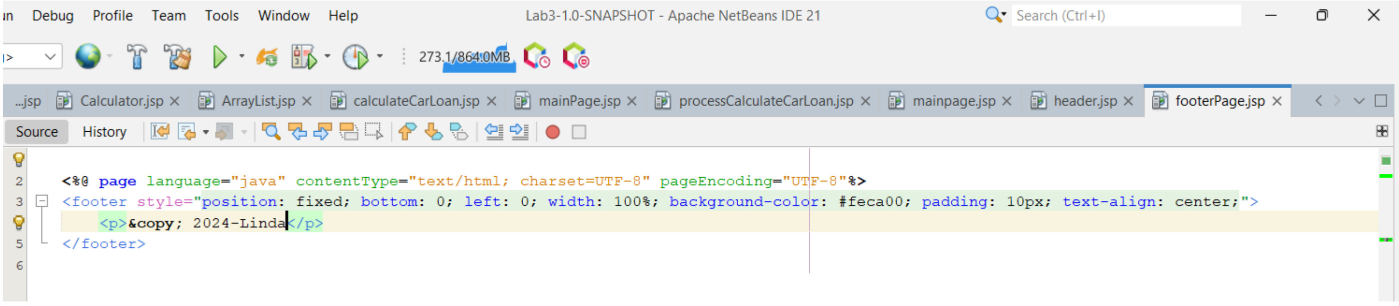
THE CODE:



```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Main Page</title>
</head>
<body>
    <%@ include file="header.jsp" %>

    <h1>Using Java Include directive</h1>
    <p style="font-size: small; color: red;">Java Server Page (JSP) is a technology
        for controlling the content or appearance of Web pages through the use of servlets,
        small programs that are specified in the Web page and run on the Web server
        to modify the Web page before it is sent to the user who requested it.</p>

    <%@ include file="footerPage.jsp" %>
</body>
</html>
```

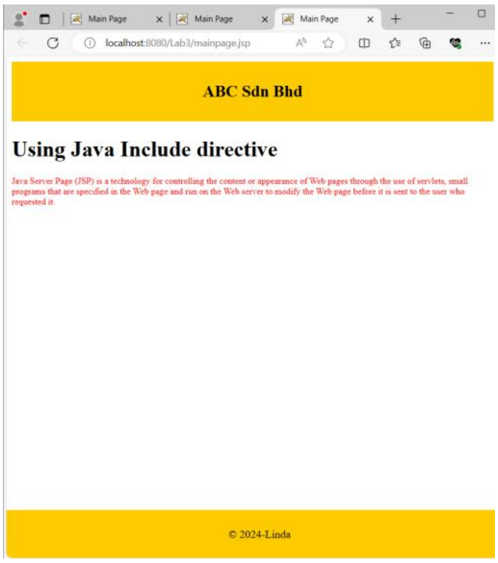


```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<div style="background-color: #feca00; padding: 10px; text-align: center;">
    <h2 style="font-size: 24px;">ABC Sdn Bhd</h2>
</div>
```



```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<footer style="position: fixed; bottom: 0; left: 0; width: 100%; background-color: #feca00; padding: 10px; text-align: center;">
    <p>&copy; 2024-Linda</p>
</footer>
```

THE OUTPUT:



ABC Sdn Bhd

Using Java Include directive

Java Server Page (JSP) is a technology for controlling the content or appearance of Web pages through the use of servlets, small programs that are specified in the Web page and run on the Web server to modify the Web page before it is sent to the user who requested it.

© 2024-Linda

Exercise

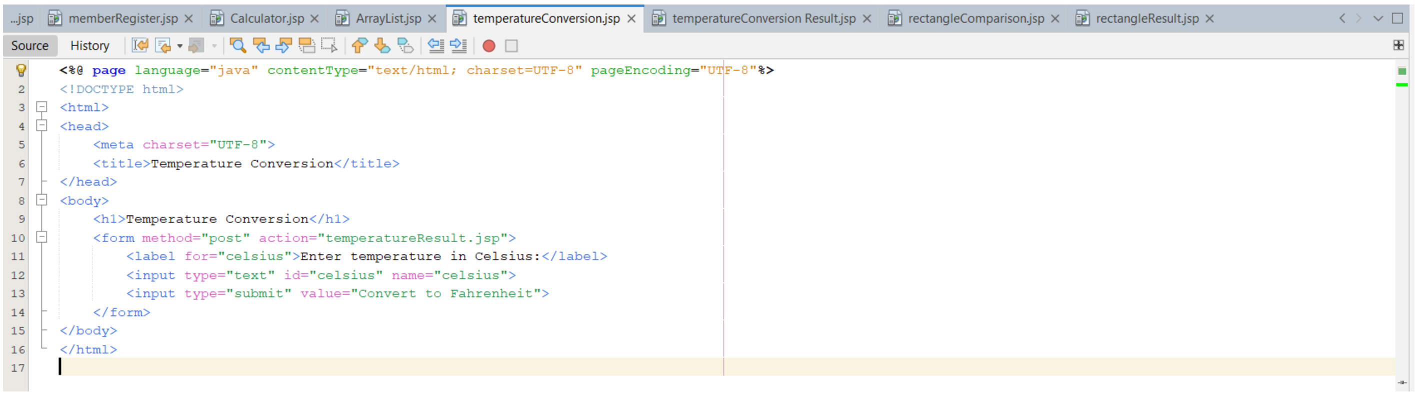
1. Write a JSP's page to convert temperatures to Fahrenheit temperatures and via versa. The formula is given as:

   $$F = (9/5)C + 32$$

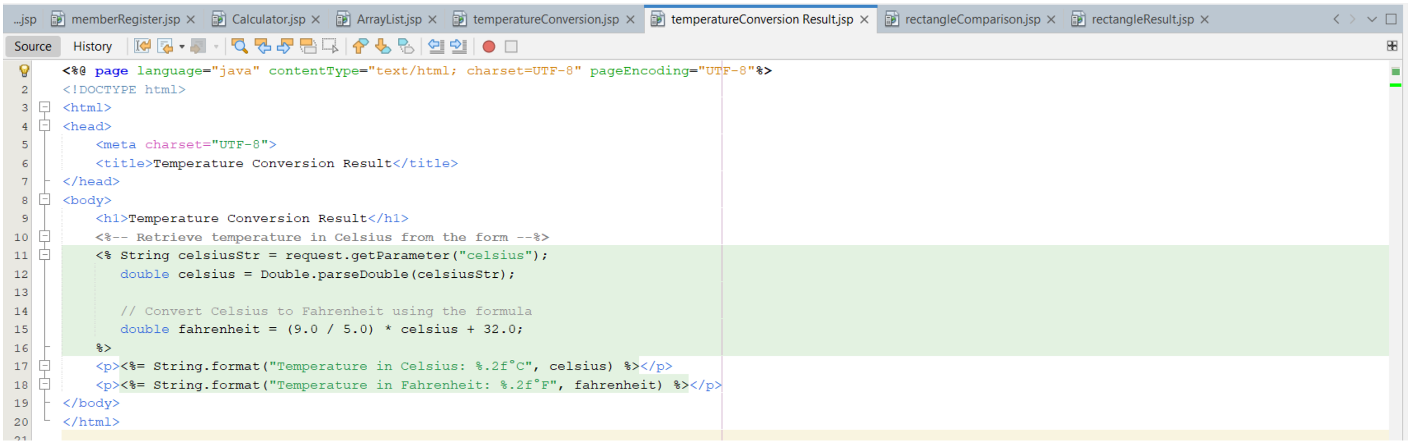
   Your program should ask the user to enter a temperature in Celsius, and then display the temperature converted to Fahrenheit.

2. Write a JSP's form that asks for the length and width of two rectangles. The program should tell the user which rectangle has the greater area, or if the areas are the same. [Note: All result must be in 2 decimal places].

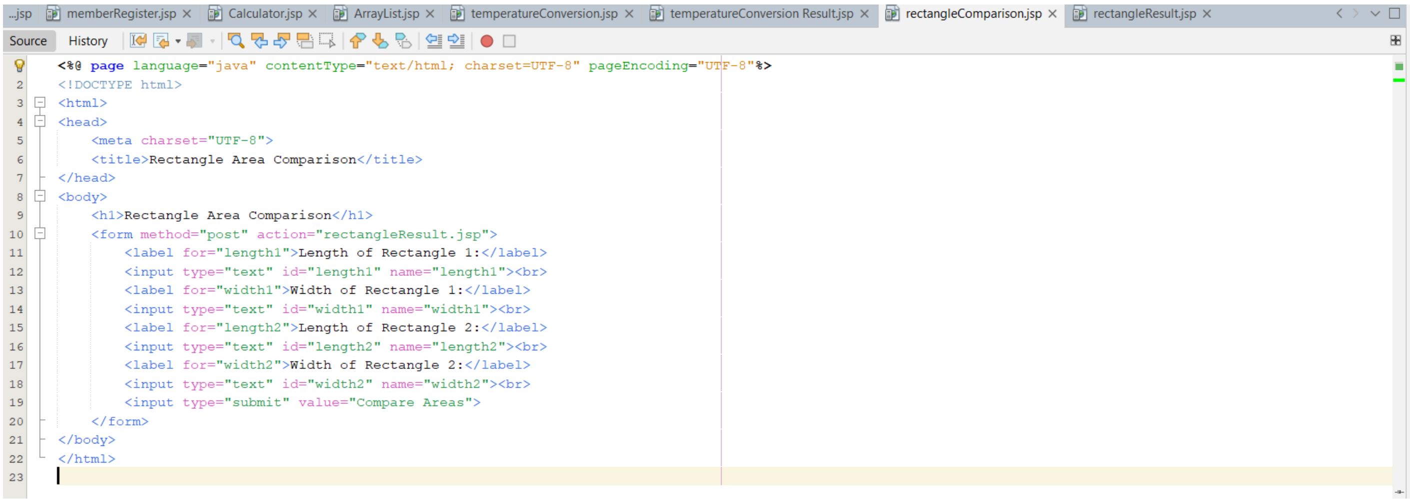
THE CODE:



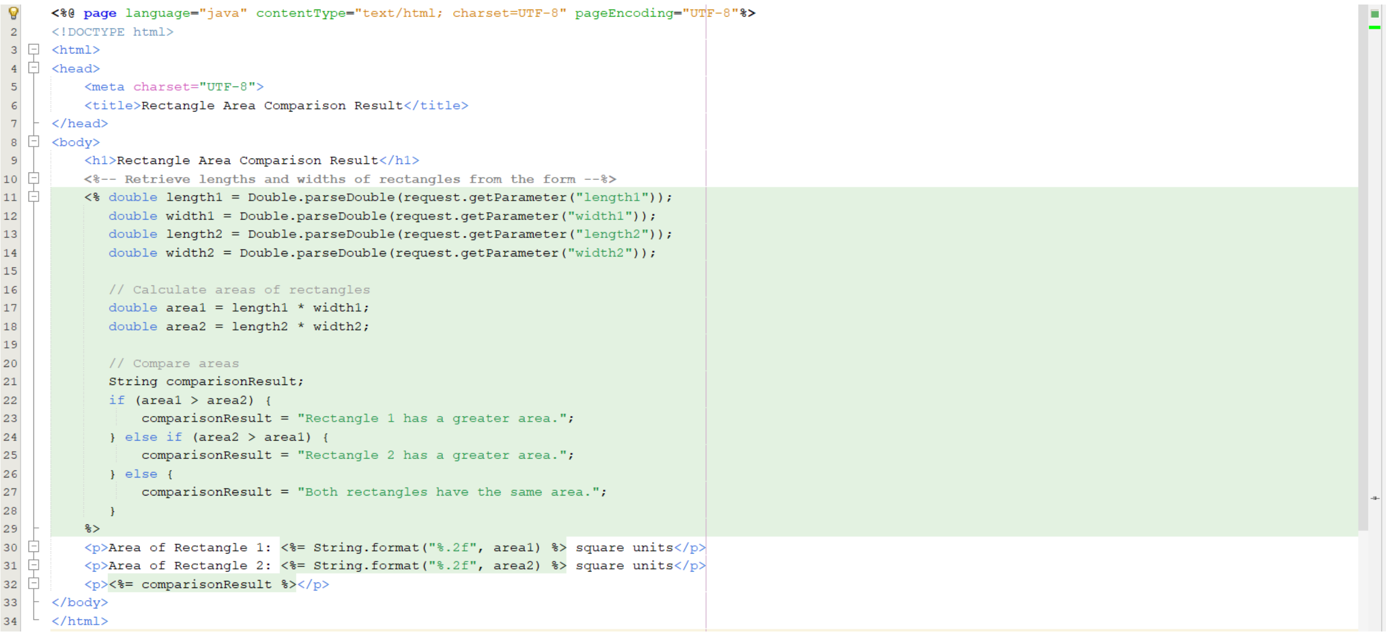
```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Temperature Conversion</title>
</head>
<body>
    <h1>Temperature Conversion</h1>
    <form method="post" action="temperatureResult.jsp">
        <label for="celsius">Enter temperature in Celsius:</label>
        <input type="text" id="celsius" name="celsius">
        <input type="submit" value="Convert to Fahrenheit">
    </form>
</body>
</html>
```



```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Temperature Conversion Result</title>
</head>
<body>
    <h1>Temperature Conversion Result</h1>
    <%-- Retrieve temperature in Celsius from the form --%>
    <% String celsiusStr = request.getParameter("celsius");
       double celsius = Double.parseDouble(celsiusStr);

       // Convert Celsius to Fahrenheit using the formula
       double fahrenheit = (9.0 / 5.0) * celsius + 32.0;
    %>
    <p><%= String.format("Temperature in Celsius: %.2f°C", celsius) %></p>
    <p><%= String.format("Temperature in Fahrenheit: %.2f°F", fahrenheit) %></p>
</body>
</html>
```

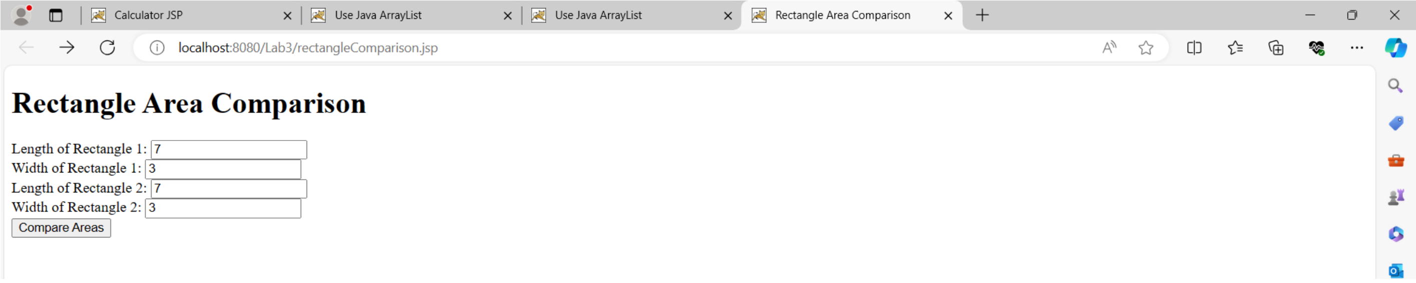


```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Rectangle Area Comparison</title>
</head>
<body>
    <h1>Rectangle Area Comparison</h1>
    <form method="post" action="rectangleResult.jsp">
        <label for="length1">Length of Rectangle 1:</label>
        <input type="text" id="length1" name="length1"><br>
        <label for="width1">Width of Rectangle 1:</label>
        <input type="text" id="width1" name="width1"><br>
        <label for="length2">Length of Rectangle 2:</label>
        <input type="text" id="length2" name="length2"><br>
        <label for="width2">Width of Rectangle 2:</label>
        <input type="text" id="width2" name="width2"><br>
        <input type="submit" value="Compare Areas">
    </form>
</body>
</html>
```



```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Rectangle Area Comparison Result</title>
</head>
<body>
    <h1>Rectangle Area Comparison Result</h1>
    <%-- Retrieve lengths and widths of rectangles from the form --%>
    <% double length1 = Double.parseDouble(request.getParameter("length1"));
       double width1 = Double.parseDouble(request.getParameter("width1"));
       double length2 = Double.parseDouble(request.getParameter("length2"));
       double width2 = Double.parseDouble(request.getParameter("width2"));

       // Calculate areas of rectangles
       double area1 = length1 * width1;
       double area2 = length2 * width2;

       // Compare areas
       String comparisonResult;
       if (area1 > area2) {
           comparisonResult = "Rectangle 1 has a greater area.";
       } else if (area2 > area1) {
           comparisonResult = "Rectangle 2 has a greater area.";
       } else {
           comparisonResult = "Both rectangles have the same area.";
       }
    %>
    <p>Area of Rectangle 1: <%= String.format("%.2f", area1) %> square units</p>
    <p>Area of Rectangle 2: <%= String.format("%.2f", area2) %> square units</p>
    <p><%= comparisonResult %></p>
</body>
</html>
```

THE OUTPUT:


Calculator JSP
Use Java ArrayList
Use Java ArrayList
Rectangle Area Comparison
localhost:8080/Lab3/rectangleComparison.jsp

Rectangle Area Comparison

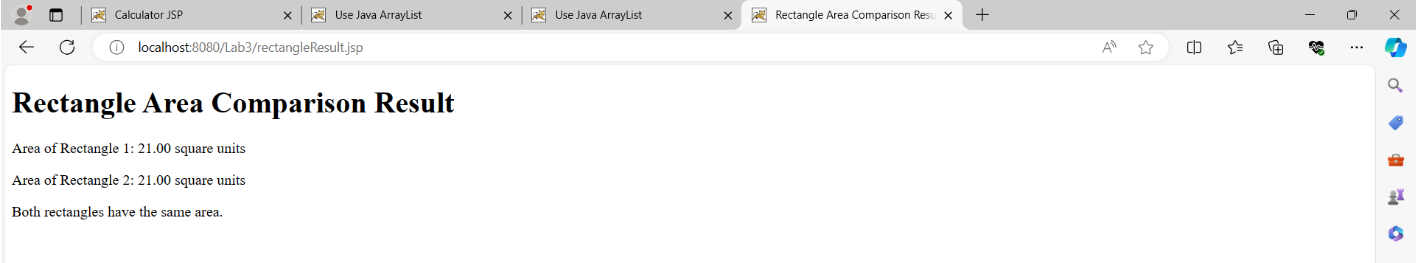
Length of Rectangle 1: 7

Width of Rectangle 1: 3

Length of Rectangle 2: 7

Width of Rectangle 2: 3

Compare Areas


Calculator JSP
Use Java ArrayList
Use Java ArrayList
Rectangle Area Comparison Resu
localhost:8080/Lab3/rectangleResult.jsp

Rectangle Area Comparison Result

Area of Rectangle 1: 21.00 square units

Area of Rectangle 2: 21.00 square units

Both rectangles have the same area.