

Week 3

JSP: Scriptlet, Expression & Standard Actions

Web Programming 2

Name: Luqman Hakim Bin Aziz

Matric #: S66292

Semester: II 2023/2024

Lab: 4

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

Task 1: Using JSP Scripting.....	11
Task 2: Using JSP (Scripting, Declaration and Expression)	16
Task 3: Using JSP Standard Action (Include and Param).....	20
Task 4: Using JSP Standard Action (Forward)	25
Task 5: Use Java Scriptlet To Construct Business Logic	30

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 tulis kesimpulan 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: Using JSP Scripting

Objective : JSP Scriptlet and JSP Expression in application.

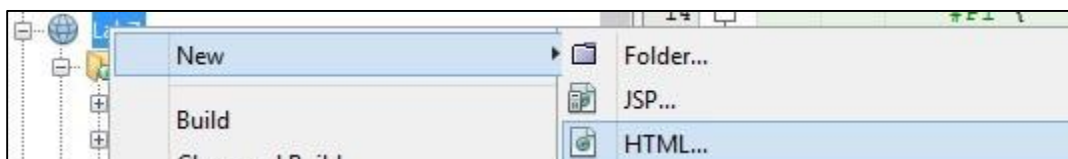
Problem : Prepare a simple interface to perform the following payment process;

Description

- i. If Customer Type is Normal Customer (assign value as “1”) and Order Quantity > 100, customer entitle 10% discount.
- ii. If Customer Type is Privilege Customer (assign value as “2”) and Order Quantity > 100, customer entitle 25% discount.
- iii. Order Quantity must be in number.
- iv. Finally, display the results.

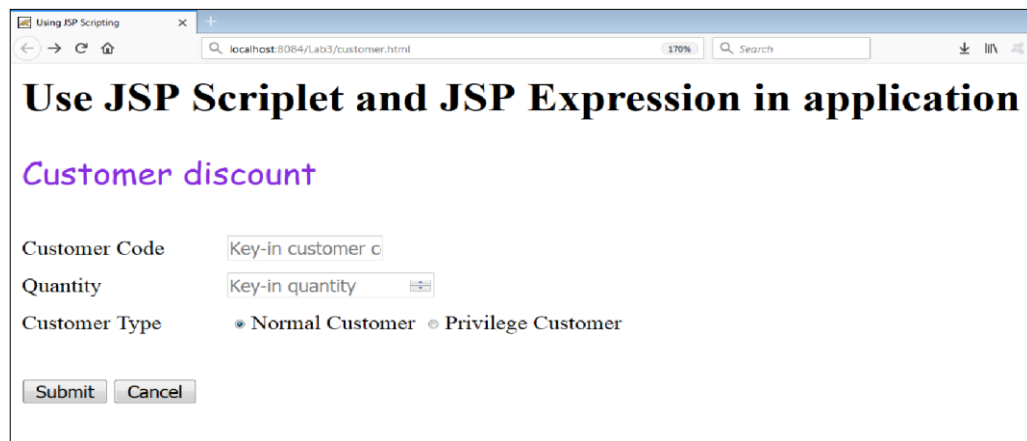
Estimated time : 40 minutes

1. Create Project *Lab3*.
2. Create a new HTML's file.



3. Type file name as *customer*.

4. Prepare the following Graphical User Interface (GUI).



Use JSP Scriptlet and JSP Expression in application

Customer discount

Customer Code

Quantity

Customer Type ☒ Normal Customer ☐ Privilege Customer

5. You must ensure the amount must be written as number.
6. The value for Normal Customer is “1” and Privilege Customer is “2”
7. Create a new file name known as *processCustomer.jsp*.
8. Define related variables and methods as below.

```
<%  
    final int price = 10;  
  
    //Using JSP Scriptlet...  
    String cust_no1 = request.getParameter("cust_no");  
    int quantity1 = Integer.parseInt(request.getParameter("quantity"));  
    String cust_type1 = request.getParameter("cust_type");  
  
    //Determine customer..  
    if (cust_type1.equals("1") && quantity1 > 100) {  
        out.print("You're entitle " + "10%");<br><%  
            out.print("Total amount is RM" + quantity1 * price * 0.9);  
        } else if (cust_type1.equals("2") && quantity1 > 100) {  
            out.print("You're entitle " + "25%");<br><%  
                out.print("Total amount is RM" + quantity1 * price * 0.75);  
            } else {  
                out.print("You're not entitle discount..!");<br><%  
                    out.print("Total amount is RM" + quantity1 * price);  
                }  
            }  
>%
```

9. Compile *customer.html* and *processCustomer.jsp* file.
10. Run *customer.html*.

11. Enter information to the interface.

12. Output will appear in web browser.

Use JSP Scriptlet and JSP Expression in application

You're entitle 10%
Total amount is RM2250.0

Coding customer.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Using JSP Scripting</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
      h2 {
        color: blue;
      }
    </style>
  </head>
  <body>
    <h1>Use JSP Scriptlet and JSP Expression in application</h1>
    <h2>Customer discount</h2>

    <form action="processCustomer.jsp" method="post">
      <label for="customercode">Customer Code :</label>
      <input type="number" id="customercode" name="customercode" placeholder="Key-in customer code"required><br><br>

      <label for="quantity">Quantity :</label>
      <input type="number" id="quantity" name="quantity" placeholder="Key-in quantity"required><br><br>

      <label>Customer Type:</label>
      <ul id="custtype" name="custtype" required>
        <input type="radio" id="1" name="custtype" value="1" required><label for="normal">Normal Customer</label>
        <input type="radio" id="2" name="custtype" value="2" required><label for="privilege">Privilege Customer</label>
      </ul>

      <button type="submit">Calculate</button>
      <button type="clear">Cancel</button>
    </form>
  </body>
</html>
```

Coding processCustomer.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Customer Discount Calculation</title>
  </head>
  <body>
    <%
      final int price = 10;

      String customerCode = request.getParameter("customercode");
      int quantity = Integer.parseInt(request.getParameter("quantity"));
      String customerType = request.getParameter("custtype");

      out.print("<h1>Customer Discount Calculation Result</h1>");

      if (customerType.equals("1") && quantity > 100){
        out.print("<p>You're entitled to a 10% discount</p>");
        out.print("<p>Total amount is RM" + quantity * price * 0.9 + "</p>");
      }
      else if (customerType.equals("2") && quantity > 100) {
        out.print("<p>You're entitled to a 25% discount</p>");
        out.print("<p>Total amount is RM" + quantity * price * 0.75 + "</p>");
      }
      else {
        out.print("<p>You're not entitled to a discount..!</p>");
        out.print("<p>Total amount is RM" + quantity * price + "</p>");
      }
    %>
  </body>
</html>
```

Reflection

1. What you have learnt from this exercise?

- Structure: The structure of the HTML form is well-defined with appropriate use of <form>, <label>, <input>, and <button> elements.
- Validation: Input fields are marked as required, ensuring that users cannot submit the form with empty values. However, server-side validation should also be implemented to handle cases where client-side validation fails or is bypassed.

2. Explain three (3) type of JSP scripting?

-Declaration Tags (<%! %>)

Declaration tags are used to declare variables, methods, and other members that will be accessible across multiple methods within the JSP page.

-Scriptlet Tags (<% %>)

Scriptlet tags allow embedding Java code directly within the HTML content of a JSP page.

- Expression Tags (<%= %>)

Expression tags are used to insert the result of a Java expression directly into the HTML content of a JSP page.

Objective : Use JSP Declaration tag, JSP Scriptlet and JSP Expression in application.

Problem

Description : Create currency conversion page to Malaysia Ringgit into US Dollar, Euro or Pound Sterling.

1 USD = RM3.92

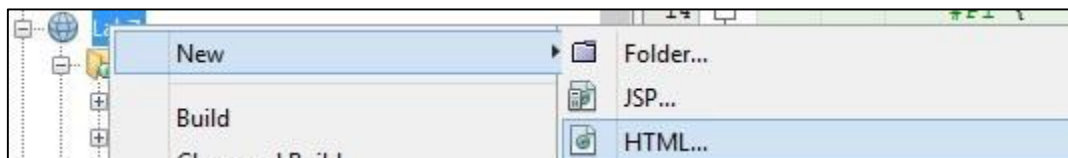
1 Pound Sterling = RM5.96

1 Euro = RM4.47;

Task 2: Using JSP (Scripting, Declaration and Expression)

Estimated time : 40 minutes

1. Choose Project *Lab3*.
2. Create a new HTML's file.



3. Type file name as *currencyConversion*.
4. Prepare the following Graphical User Interface (GUI).

Use JSP Declaration tag, JSP Scriptlet and JSP Expression in application

Currency Conversion

Amount (in RM)

Convert to

©2016-Mohamad Nor

5. You must ensure the amount must be written as number.

6. The value for USD is "1", Pound Sterling is "2" and Euro is "3"
7. Create a new file name known as *processCurrency.jsp*.
8. Define related variables, currency rate as a constant and method *calculateRate(String code, int amount)* in JSP declaration tag as below.

```
17 <!--
18 //Added by : 10 April 2016 - Mohamad Nor
19 //Define constant....
20 final double USD = 3.92;
21 final double STG = 5.96;
22 final double EURO = 4.47;
23
24 //Define method to perform currency exchange....
25 private double calculateRate(String currency, int amount)
26 {
27     double currencyChange=0.00f;
28
29     if ( currency.equals("1") )
30         currencyChange = (double)( amount * USD);
31     if ( currency.equals("2") )
32         currencyChange = (double)( amount * STG);
33     if ( currency.equals("3") )
34         currencyChange = (double)( amount * EURO);
35
36     return currencyChange; //return the result....
37 }
38 -->
```

9. In your JSP scriptlet, retrieve the value for *Amount* and *Convert to* and assign to respective variables.
10. Call method *calculateRate(String code, int amount)* to perform currency conversion.
11. Finally, display the result using JSP Expression tag.
12. Compile *currencyConversion.html* and *processCurrency.jsp* file.
13. Run *currencyConversion.html*.

14. Enter the following information

Use JSP Declaration tag, JSP Scriptlet and JSP Expression in application
Currency Conversion
Amount (in RM)
Convert to

©2016-Mohamad Nor

15. Output will appear in web browser (*Note: Amount must be in 2 decimal places*).

Use JSP Declaration tag, JSP Scriptlet and JSP Expression in application
Amount in Ringgit Malaysia is RM1000
Amount in Euro is RM223.71
©2016-Mohamad Nor

Coding *currencyConversion.html*

```
<!DOCTYPE html>
<html>
  <head>
    <title>Task 2</title>
    <style>
      h2 {
        color: purple;
      }
      h1 {
        color: black;
      }
    </style>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>

    <h1>Use JSP Scriptlet and JSP Expression in application</h1>
    <h2>Currency Conversion</h2>

    <form action="processCurrency.jsp" method="post">
      <label for="amount">Amount (RM) :</label>
      <input type="number" id="amount" name="amount" placeholder="RM" required><br><br>

      <label for="convert">Convert to:</label>
      <select id="convert" name="convert" required>
        <option value="1">USD</option>
        <option value="2">STG</option>
        <option value="3">Euro</option>

        <button type="submit">Submit</button>
        <button type="clear">Cancel</button>
      </select><br><br>

      <button type="submit">Submit</button>
      <button type="clear">Cancel</button>
      <br><br>
    </form>

  </body>
  <footer>
    <script>2024 Lucman Hakim
  </script>
</footer>
</html>
```

Coding *processCurrency.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>
<%!
    final double USD = 3.92;
    final double STG = 5.96;
    final double EURO = 4.47;

    private double calculateRate(String currency, int amount) {
        double currencyChange = 0.00;

        if (currency.equals("1"))
            currencyChange = (double) (amount / USD);
        else if (currency.equals("2"))
            currencyChange = (double) (amount / STG);
        else if (currency.equals("3"))
            currencyChange = (double) (amount / EURO);

        return currencyChange;
    }
%>

<%
    int amount = Integer.parseInt(request.getParameter("amount"));
    String convertTo = request.getParameter("convert");

    double result = calculateRate(convertTo, amount);
%>

<h1>Use JSP Scriptlet and JSP Expression in application</h1>
<p style="color: purple;">Amount in Ringgit Malaysia: RM<%= amount %></p>
<% if (convertTo.equals("1")) { %>
    <p style="color: purple;">Amount in US Dollar: <%= String.format("%.2f", result) %></p>
<% } else if (convertTo.equals("2")) { %>
    <p style="color: purple;">Amount in British Pound: <%= String.format("%.2f", result) %></p>
<% } else if (convertTo.equals("3")) { %>
    <p style="color: purple;">Amount in Euro: <%= String.format("%.2f", result) %></p>
<% } %>

<footer>
    <small>copy:2024 Lugman Hakim
</small>
</footer>
</body>
</html>

```

Reflection

1. What have you learn from this exercise?

- JSP (JavaServer Pages): I've seen how JSP allows for the creation of dynamic web pages by embedding Java code within HTML. This enables the generation of content dynamically based on user input or other factors.

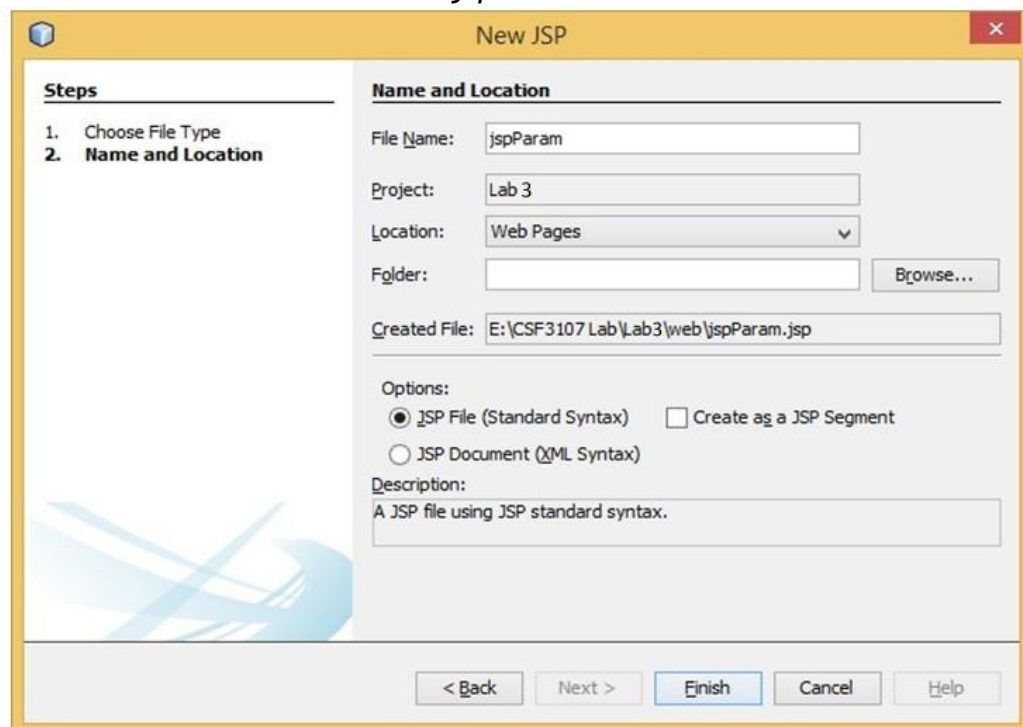
Objective : Using `<jsp:include>` and `<jsp:param>` to display information on JSP page

Problem Description : Display the course information.

Estimated time : 20 minutes

1. Go to Project *Lab3*.

2. Create a new JSP's file known as *jspParameter*.



3. Prepare the following HTML's syntax.

```
1 <!--
2     Document    : jspParam
3     Created on  : 11-Apr-2016, 14:06:19
4     Author     : Mohamad Nor Hassnan
5 -->
6
7 <%@page contentType="text/html" pageEncoding="UTF-8"%>
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
12     <title>Using JSP Standard Action </title>
13 </head>
14 <body>
15     <h1>Using jsp:include and jsp:param to display information on JSP page</h1>
16 </body>
17 </html>
```

4. Add Java scriptlet.

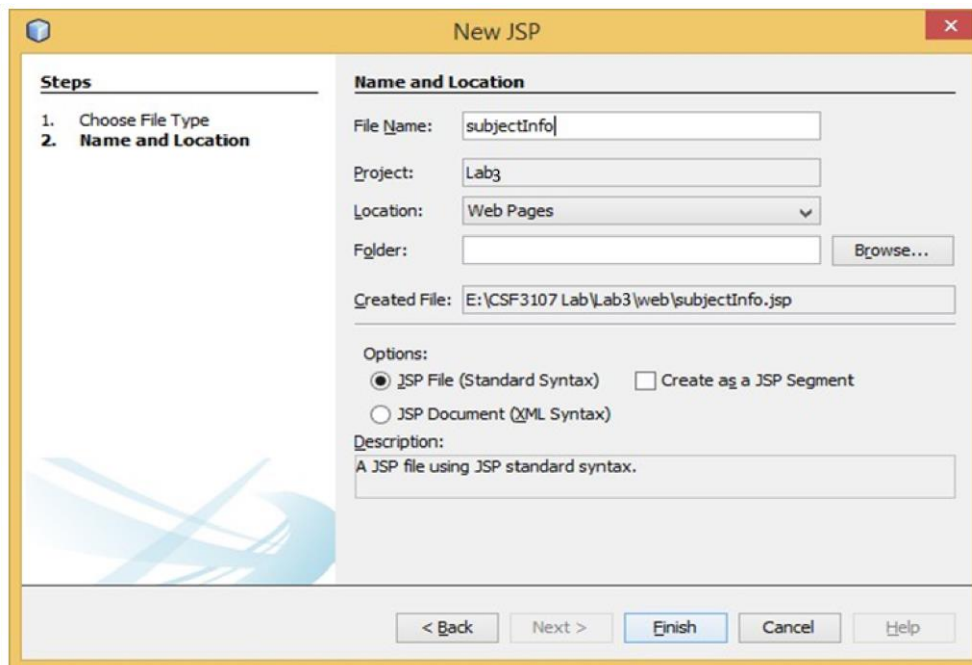
```
14 <body>
15     <h1>Using jsp:include and jsp:param to display information on JSP page</h1>
16 <%
17     String sCode = "CSF3107";
18     String sSubject = "Web Programming 2";
19     String sCredit = "3(2+1)";
20 %>
21 </body>
```

5. Add JSP Standard Action `<jsp:include>` to call *subjectInfo.jsp*'s page and `<jsp:parameter>` to store the subject's information .

```
21
22 <!-- Call subjectInfo.jsp page & passing course information
23      to respective parameters...-->
24 <jsp:include page="subjectInfo.jsp" flush="true">
25     <jsp:param name="code" value="<%=sCode%>"/>
26     <jsp:param name="subject" value="<%=sSubject%>"/>
27     <jsp:param name="credit" value="<%=sCredit%>"/>
28 </jsp:include>
29 </body>
```

6. Save *jspParameter.jsp*'s file.

7. Create another JSP's file known as *subjectInfo*.



8. Write the following HTML's syntax.

```

1  <!--
2      Document    : subjectInfo
3      Created on  : 11-Apr-2016, 14:45:36
4      Author     : Mohamad Nor
5  -->
6
7  <%@page contentType="text/html" pageEncoding="UTF-8"%>
8  <!DOCTYPE html>
9  <html>
10     <head>
11         <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
12         <title>Using JSP Standard Action</title>
13     </head>
14     <body>
15         <h1>Calling subjectInfo.jsp page</h1>
16     </body>
17 </html>

```

9. Add three (3) paragraphs and use JSP expression to retrieve and assign value to these paragraphs.

```

14     <body>
15         <h1>Calling subjectInfo.jsp page</h1>
16         <p>Code = <%=request.getParameter("code")%></p>
17         <p>Subject = <%=request.getParameter("subject")%></p>
18         <p>Credit = <%=request.getParameter("credit")%></p>
19     </body>

```

11. Save all files.

12. Compile and run *jspParameter.jsp*'s file.

13. Output will appear in web browser.

jspParameter.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Using JSP Standard Action</title>
    </head>
    <body>
        <h1>Using jsp:include and jsp:param to display information on JSP page</h1>
        <%
            String sCode = "CSM3023";
            String sSubject = "Web Programming 2";
            String sCredit = "3 (2+1) ";
        %>

        <jsp:include page="subjectInfo.jsp" flush="true">
            <jsp:param name="code" value="<%=sCode%>" />
            <jsp:param name="subject" value="<%=sSubject%>" />
            <jsp:param name="credit" value="<%=sCredit%>" />
        </jsp:include>
    </body>
</html>
```

subjectInfo.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Using JSP Standard Action</title>
    </head>
    <body>
        <h1>Calling subjectInfo.jsp</h1>
        <p>Code = <%=request.getParameter("code")%></p>
        <p>Subject = <%=request.getParameter("subject")%></p>
        <p>Credit = <%=request.getParameter("credit")%></p>
    </body>
</html>
```

Reflection

1. What you have learnt from this exercise?

- This exercise demonstrated the usage of JSP standard actions such as `jsp:include` and `jsp:param` to include one JSP page within another and pass parameters between them.

2. List **TWO (2)** other JSP Standard Action Tag.

- `jsp:forward`: This action forwards the request from one JSP page to another resource (such as another JSP page, servlet, or HTML file) on the server.
- `jsp:useBean`: This action is used to instantiate a JavaBean component or retrieve an existing one from the request, session, or application scope. It simplifies the process of working with JavaBeans in JSP pages, allowing for easier access to data and logic encapsulated within Java classes.

Task 4: Using JSP Standard Action (Forward)

Objective : Using `<jsp: forward>` to display user information and object on JSP page

Problem : Display user information.

Estimated time : 20 minutes

1. Go to Project *Lab3*.

2. Create a new JSP's file known as *forward*.

Steps

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

Name and Location

File Name:

Project:

Location:

Folder:

Created File:

Options:

☒ JSP File (Standard Syntax) ☐ Create as a JSP Segment

☐ JSP Document (XML Syntax)

Description:

< Back Next > **Finish** Cancel Help

3. Prepare the following HTML's syntax.

```
<html>
<head>
<title>Using JSP Standard Action(forward)</title>
</head>
<body>
<h2>Using jsp:forward to display user info.</h2>
</body>
</html>
```

4. Add JSP Standard Action `<jsp:forward>` to call `forwardInfo.jsp`'s page and `<jsp:parameter>` to store the user's information.

```

<body>
    <h2>Using jsp:forward to display user info.</h2>
    <jsp:forward page="forwardInfo.jsp">
        <jsp:param name="U_Name" value="Fouad Abdulameer"/>
        <jsp:param name="Email" value="fouadaug@gmail.com"/>
        <jsp:param name="Nationality" value="Iraqi"/>
        <jsp:param name="Background" value="Developer"/>
    </jsp:forward>
</body>

```

5. Save *forward.jsp*'s file.
6. Create another JSP's file known as *forwardInfo*.

New JSP

Steps

1. Choose File Type
2. Name and Location

Name and Location

File Name:

Project:

Location:

Folder:

Created File:

Options:

☒ JSP File (Standard Syntax) ☐ Create as a JSP Segment

☐ JSP Document (XML Syntax)

Description:

< Back Next > **Finish** Cancel Help

7. Write the following code.

```

<html>
<head>
<title>&lt;Forwarded_Action Example in JSP&gt;</title>
</head>
<body>

<% String name = request.getParameter("U_Name"); %>
<% String Email = request.getParameter("Email"); %>
<% String Nationality = request.getParameter("Nationality"); %>
<% String Background = request.getParameter("Background"); %>
<% if (name != null) { %>
<b><br><br><h2 align="center">

    <%=name%><br>
    <%=Email%><br>
    <%=Nationality%><br>
    <%=Background%><br><br>

    <% out.print("Today is:" + java.util.Calendar.getInstance().getTime()); %>
</h2></b></br>
<%= %>

</body>
</html>

```

8. Save all files.

9. Compile and run *forward.jsp*'s file.

10. Output will appear in web browser.

Coding forward.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Using JSP Standard Action(forward)</title>
</head>
<body>
<h2>Using jsp:forward to display user info.</h2>
<jsp:forward page="forwardInfo.jsp">
    <jsp:param name="U_Name" value="Luqman Hakim"/>
    <jsp:param name="Email" value="loqmanhakeem999@gmail.com"/>
    <jsp:param name="Nationality" value="Malaysia"/>
    <jsp:param name="Background" value="Developer"/>
</jsp:forward>
</body>
</html>

```

Coding forwardInfo.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>&lt;Forwarded_Action Example in JSP&gt;</title>
  </head>
  <body>

    <% String name = request.getParameter("U_Name"); %>
    <% String Email = request.getParameter("Email"); %>
    <% String Nationality = request.getParameter("Nationality"); %>
    <% String Background = request.getParameter("Background"); %>
    <% if (name != null) {%>
      <b><br><br><h2 align="center">

        <%=name%><br>
        <%=Email%><br>
        <%=Nationality%><br>
        <%=Background%><br><br>

        <% out.print("Today is:" + java.util.Calendar.getInstance().getTime());%>
      </h2></b></br>
    <%}%>

  </body>
</html>
```

Reflection

1. What you have learnt from this exercise?

- forwardInfo.jsp page seems to be correctly receiving the forwarded parameters and displaying them

2. List **TWO(2)** More JSP Standard Action Tag.

- jsp:include: This tag is used to include the content of another resource (like another JSP page, HTML file, or servlet) into the current JSP page during the translation phase. It's similar to the `<%@include %>` directive, but it provides more flexibility at runtime.
- jsp:useBean: This tag is used to declare and instantiate JavaBeans (Java objects) for use within the JSP page. It allows you to use Java objects to encapsulate data and logic separately from the presentation layer.

Task 5: Use Java Scriptlet To Construct Business Logic

Objective : Use Java Scriptlet to perform business logic.

Problem Description : Create a simple web based form to calculate the insurance quotation.

Coverage type - Third Party (value as "1")

Comprehensive ((value as "2"))

Formula for insurance comprehensive

NCD = 55%, 1.8% x market price

NCD = 35%, 2.4% x market price

NCD = 25%, 3.0% x market price

NCD = 10%, 3.8% x market price

Formula for third party

NCD = 55%, 1.2% x market price

NCD = 35%, 1.8% x market price

NCD = 25%, 2.5% x market price

NCD = 10%, 3.3% x marketprice

Estimated time : 50 minutes

1. Go to project *Lab3*.
2. Create a new JSP's file as *insuranceQuotation*.

3. Prepare the following Graphical User Interface (GUI).

Insurance Quotation

Insurance Calculation

ICNo *

E.g. 821210-05-3478

Name *

Enter name

Market Price *

Price

Coverage Type

Third Party ▼

No claims discount (NCD)

10% ▼
10%
25%
35%
55%

Submit

Cancel

©2016-Mohamad Nor

4. You need to ensure all front-end validation take place.
5. Creating another JSP's file known as *processInsuranceQuo.jsp*.
6. Use Java Scriplet to perform the business logic for the application in page *processInsuranceQuo.jsp*.
7. Final insurance amount must be added with 6% GST.
8. Save your file.
9. Right click *insuranceQuotation.jsp* and compile the program.

10. Test your application by key-in the following information.

Insurance Quotation

Insurance Calculation

ICNo*

870510-11-2167

Name*

Mohd Razali Abdullah

Market Price*

40000

Coverage Type

Comprehensive ▼

No claims discount (NCD)

35% ▼

Submit

Cancel

©2016-Mohamad Nor

11. You should get the following output.

Details of Insurance Quotation

IC No : 870510-11-2167

Customer Name : Mohd Razali Abdullah

Market Price : 40000

Coverage Type : Comprehensive

No claim discount (NCD) = 35%

Insurance amount : 960.00

6% GST : 57.60

Final amount (with 6% GST) : 1017.60

insuranceQuotation.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<title>Insurance Quotation</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<script>
function validateForm() {
    let icno = document.getElementById("icno").value;
    let name = document.getElementById("name").value;
    let price = document.getElementById("price").value;

    if(icno === "" || name === "" || price === "") {
        alert("Please fill in all fields");
        return false;
    }
    return true;
}
</script>
</head>
<body>
<h1>Insurance Quotation</h1>

<form action="processInsuranceQuo.jsp" onsubmit="return validateForm()">
    <fieldset>
        <legend>Insurance Calculation</legend>
        <label for="icno">ICNo :</label>
        <input type="number" id="icno" name="icno"><br><br>
        <label for="name">Name :</label>
        <input type="text" id="name" name="name"><br><br>
        <label for="price">Market Price :</label>
        <input type="number" id="price" name="price"><br><br>
        <label for="cover">Coverage Type:</label>
        <select name="cover" id="cover">
            <option value="1" selected="selected">Comprehensive</option>
            <option value="2">Third-party</option>
        </select><br><br>
        <label for="disc">No claims discount (NCD):</label>
        <select name="disc" id="disc">
            <option value="1" selected="selected">10%</option>
            <option value="2">25%</option>
            <option value="3">35%</option>
            <option value="4">55%</option>
        </select><br><br>
        <button type="submit">Calculate</button>
        <button type="reset">Clear</button>
    </fieldset>
</form>
</body>
</html>
```

processInsuranceQuo.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%
    String icNo = request.getParameter("icno");
    String customerName = request.getParameter("name");
    double marketPrice = Double.parseDouble(request.getParameter("price"));
    String coverageType = request.getParameter("cover");
    int ncd = Integer.parseInt(request.getParameter("disc"));

    double discount = 0.0;
    switch (ncd) {
        case 1:
            discount = 0.1;
            break;
        case 2:
            discount = 0.25;
            break;
        case 3:
            discount = 0.35;
            break;
        case 4:
            discount = 0.55;
            break;
    }

    double totalAmount = marketPrice * (1 - discount);
    double gst = totalAmount * 0.06;
    double finalAmount = totalAmount + gst;
%>

<html>
<body>
    <h1>Insurance Quotation Result</h1>
    <p>IC No: <%= icNo %></p>
    <p>Customer Name: <%= customerName %></p>
    <p>Market Price: <%= marketPrice %></p>
    <p>Coverage Type: <%= (coverageType.equals("1")) ? "Comprehensive" : "Third-party" %></p>
    <p>No Claim Discount (NCD): <%= (discount * 100) %>%</p>
    <p>Insurance Amount: <%= totalAmount %></p>
    <p>6% GST: <%= gst %></p>
    <p><strong>Final Amount (with 6% GST): <%= finalAmount %></strong></p>
</body>
</html>
```

Reflection

1. What you have learnt from this exercise?

- code processes the data submitted from the form you provided and calculates the insurance quotation

2. List all Java features you used in Java Scriptlet.

- Variable Declaration and Assignment: Java allows the declaration and assignment of variables to store data. For example:

```
String icNo = request.getParameter("icno");
```

```
double marketPrice = Double.parseDouble(request.getParameter("price"));
```

- Conditional Statements (Switch-case): The switch-case statement is used to select one of many code blocks to be executed.

- Arithmetic Operations: Java supports various arithmetic operations for performing calculations. For example:

Calculating the total amount: `double totalAmount = marketPrice * (1 - discount);`

Calculating GST: `double gst = totalAmount * 0.06;`

- String Manipulation: Java provides methods for manipulating strings. For example:

`String icNo = request.getParameter("icno");`

`String coverageType = request.getParameter("cover");`

- Type Conversion: Java allows converting one data type to another. For example:

Parsing a string to a double: `Double.parseDouble(request.getParameter("price"));`

Converting an integer to a string: `(discount * 100) + "%"`

Exercise

1. Write a simple application to calculate and display a person's body mass index (BMI). The BMI is often used to determine whether a person is overweight or underweight for his or her height. A person's BMI is calculated with the following formula:

$$\text{BMI} = \text{weight} / \text{height}^2$$

where weight is measured in kilogram and height is measured in meter. User should enter his or her weight and height and then display the user's BMI.

The program should also display a message indicating whether the person has optimal weight, is underweight, or is overweight. A person's weight is considered to be optimal if his or her BMI is between 18.5 and 25. If the BMI is less than 18.5, the person is considered to be underweight. If the BMI value is greater than 25, the person is considered to be overweight.

Coding bmiCalculator.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html>
<head>
    <title>BMI Calculator</title>
</head>
<body>
    <h1>BMI Calculator</h1>
    <form method="post" action="calculateBMI.jsp">
        Enter your weight in kilograms: <input type="text" name="weight"><br>
        Enter your height in meters: <input type="text" name="height"><br>
        <input type="submit" value="Calculate BMI">
    </form>
</body>
</html>
```

Coding calculateBMI.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ page import="java.text.DecimalFormat" %>
<%@ page import="java.io.PrintWriter" %>

<%
    double weight = Double.parseDouble(request.getParameter("weight"));
    double height = Double.parseDouble(request.getParameter("height"));
    double bmi = weight / (height * height);

    DecimalFormat df = new DecimalFormat("#.##");
    String bmiFormatted = df.format(bmi);

    String bmiCategory;
    if (bmi < 18.5) {
        bmiCategory = "underweight";
    } else if (bmi >= 18.5 && bmi <= 25) {
        bmiCategory = "optimal weight";
    } else {
        bmiCategory = "overweight";
    }
%>

<html>
    <head>
        <title>BMI Result</title>
    </head>
    <body>
        <h1>Your BMI Result</h1>
        <p>Your BMI is: <%= bmiFormatted %></p>
        <p>You are <%= bmiCategory %>.</p>
    </body>
</html>
```

BMI Calculator

Enter your weight in kilograms:

Enter your height in meters:

Your BMI Result

Your BMI is: 24.22

You are optimal weight.