
Date 14/5/2023

CSM3023

WEB BASED APPLICATION DEVELOPMENT

DR FAIZAH BINTI APLOP

OMAR ISMAIL ABDJALEEL ALOMORY

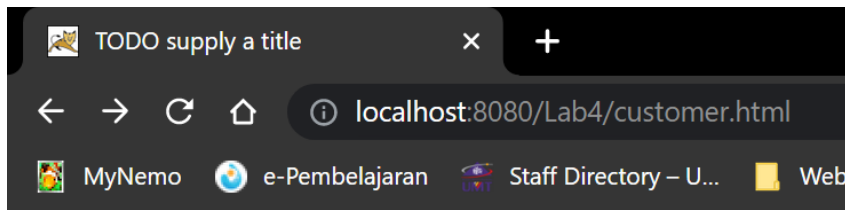
LAB 3(MP2)

Task 1: Using JSP Scripting

Compiled successfully (processCusotmer.jsp).

```
ant -f "C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4" -Dmb.internal.action.name=compile.single -Dfor.compile.single-jsp:
init:
deps-module-jar:
deps-ear-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
Created dir: C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\src
Created dir: C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\classes
Compiling 1 source file to C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\classes
BUILD SUCCESSFUL (total time: 8 seconds)
```

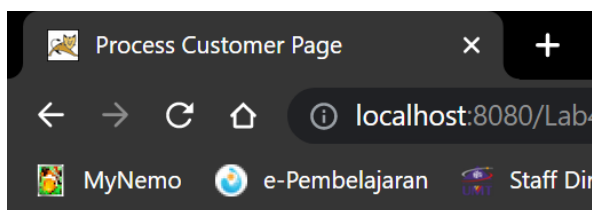
Output.



Use JSP Scriptlet and JSP Express

Customer discount

Customer Code
Quantity
Customer Type ☐ Normal Customer ☐ Privilege Customer



You're entitle10%
Total amount is RM1080.0

Reflection:

1. What you have learnt from this exercise?

We have learnt how to use jsp scriptlet to deal with client request and process the logic (if, else, etc), then display the required info back to the client.

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

1. Scriptlet:

```
<%  
    String message = "Hello, World!";  
    out.println(message);  
%>
```

2. Declaration:

```
<%!  
    private String name = "John Doe";  
    public void greet() {  
        out.println("Hello, " + name + "!");  
    }  
%>
```

3. Expression:

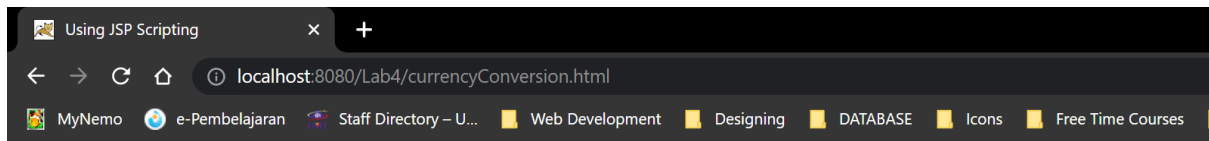
```
<p>Welcome, <%= request.getParameter("username") %>!</p>
```

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

Compiled Successfully.



Output.

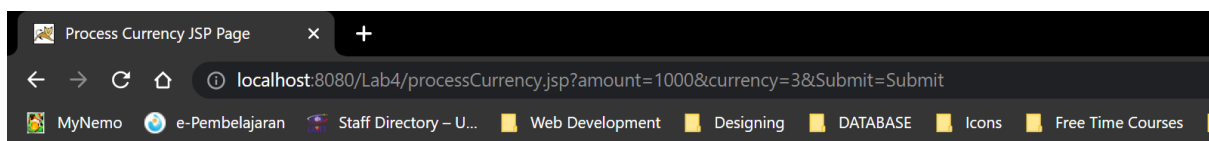


Use JSP Declaration tag, JSP Scriptlet and JSP Expression in application

Currency Conversion

Amount (in RM)

Convert to



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

Amount in Ringgit Malaysia is RM1000
Amount in Euro is RM223.71

Reflection:

1. What have you learn from this exercise?

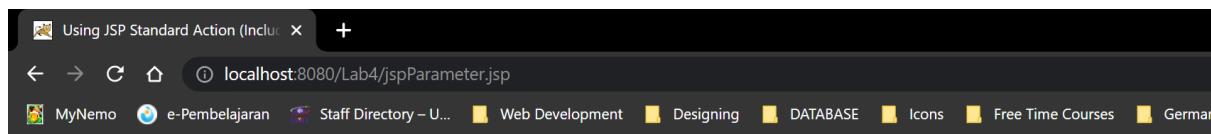
I have learned how to get the client information from HTML file to JSP file, then processing the information (do calculation) using JSP Declaration to declare method for processing request parameters, then using JSP Scriptlet to display and format the output, finally, use JSP Expression to assign the result to the output dynamically.

Task 3: Using JSP Standard Action (Include and Param)

Compiled Successfully.

```
ant -f "C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4" -Dmb.internal.action.name=compile.single -Dfo:
compile-single-jsp:
init:
deps-module-jar:
deps-war-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
Compiling 1 source file to C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\classes
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output.



Using `jsp:include` and `jsp:param` to display information on JSP page Calling `subjectInfo.jsp` page

Code = CSM 3023

Subject = Web Programming

Credit = 3(2+1)

Reflection:

1. What you have learnt from this exercise?

I have learnt how to use JSP Standard Action (Include and Param). First we initialized some parameters in the first page, then we used JSP Standard Action (include) to include these parameters in the second page so that we can call them, we attach the parameters to the next page by calling `<jsp:include page="subjectInfo.jsp" flush="true">`, then we assign each value to its variable `<jsp:param name="code" value="<%=sCode%>" />`, etc. Finally we closed the include tag `</jsp:include>`.

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

- `<jsp:forward>`: The `<jsp:forward>` tag is used to forward the request to another resource, such as another JSP page, servlet, or HTML file. It allows you to redirect the control from one resource to another, and the response is sent back to the client as if the client had requested the forwarded resource directly.

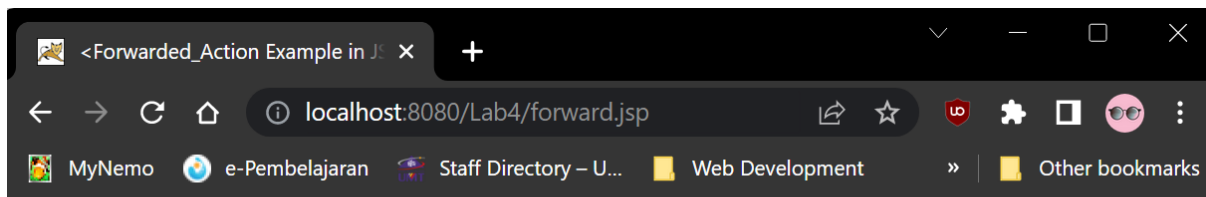
- `<jsp:useBean>`: The `<jsp:useBean>` tag is used to instantiate and manage JavaBeans within a JSP page. JavaBeans are reusable Java components that encapsulate data and behavior. The `<jsp:useBean>` tag allows you to create an instance of a JavaBean or retrieve an existing instance from a scope (such as request, session, or application scope) and make it available for use within the JSP page.

Task 4: Using JSP Standard Action (Forward)

Compiled Successfully.

```
ant -f "C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4" -Dmb.internal.action.name=compile.single -Dfor
compile-single-jsp:
init:
deps-module-jar:
deps-eaz-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
Compiling 1 source file to C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\classes
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output.



Fouad Abdulameer
fouadaug@gmail.com
Iraqi
Developer

Today is:Sun May 14 19:28:38 SGT 2023

Reflection:

1. What you have learnt from this exercise?

I have learnt how to use JSP action tag `jsp:forward` to forward information from a page to another. The syntax is as include tag. But here we are terminating the current stream flow to another JSP page with the specified parameters.

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

- `<jsp: setProperty>`: set the value of property in beans object.
- `<jsp: getProperty>`: print the value of property in beans object.
- `<jsp: param>`: sets the parameter value. It is used in forward and include mostly.

Task 5: Use Java Scriptlet to Construct Business Logic.

Compiled Successfully.

```
ant -f "C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4" -Dmb.internal.action.name=compile.single -Df:
compile-single-jsp:
init:
deps-module-jar:
deps-ear-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
Compiling 1 source file to C:\Users\komar\OneDrive - umt.edu.my\Semester 4\CSM3023 WEB BASED APPLICATION DEVELOPMENT by FAIZAH BINTI APLOP APLOP\Lab solution\Lab4\build\generated\classes
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output.

Insurance Quotation

Insurance Calculation	
ICNo	<input type="text" value="870510-11-2167"/>
Name	<input type="text" value="Mohd Razail Abdullah"/>
Market Price	<input type="text" value="40000"/>
Coverage Type	<input type="text" value="Comprehensive"/>
No claims discount (claim)	<input type="text" value="35%"/>
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

Details of Insurance Quotation

IC No : 870510-11-2167

Customer Name : Mohd Razail 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

Reflection:

1. What you have learnt from this exercise?

In this task, I have learnt how to use all JSP Scripting Elements to produce dynamic output to the client based on choices he/she chooses. The output will always change if the input parameters changed.

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

Expression

```
<p>Insurance amount : <%= String.format("%.2f",  
    (calculateRate(covType, cDis, pricel)))%><br/></p>  
<p>6% GST : <%=String.format("%.2f", gst)%><br/></p>  
<p>Final amount (with 6% GST) : <%= String.format("%.2f", total)%></p>  
  
<p>IC No : <%=icNo%></p>  
<p>Customer Name : <%=name%><br/></p>  
<p>Market Price : <%=pricel%><br/></p>  
<p>Coverage Type : <%=covType%><br/>
```

Declaration

```
<%!  
    //Define method to calculate the insurance quotation. ...  
    private double calculateRate(String coverageType, String cDiscount, int price) {  
        double insurance = 0.00f;  
        if (coverageType.equals("1")) {  
            if (cDiscount.equals("1"))  
                insurance = (double) (0.033 * price);  
            if (cDiscount.equals("2"))  
                insurance = (double) (0.025 * price);  
            if (cDiscount.equals("3"))  
                insurance = (double) (0.018 * price);  
            if (cDiscount.equals("4"))  
                insurance = (double) (0.012 * price);  
        } else {  
            if (cDiscount.equals("1"))  
                insurance = (double) (0.038 * price);  
            if (cDiscount.equals("2"))  
                insurance = (double) (0.03 * price);  
            if (cDiscount.equals("3"))  
                insurance = (double) (0.024 * price);  
            if (cDiscount.equals("4"))  
                insurance = (double) (0.018 * price);  
        }  
        return insurance; //return the result...  
    }  
<%>
```

Scriptlet.

```
<%  
    String icNo = request.getParameter("ic");  
    String name = request.getParameter("name");  
    int pricel = Integer.parseInt(request.getParameter("price"));  
    String covType = request.getParameter("coverageType").equals("1") ? "Third Party":"Comprehensive";  
    String cDis = request.getParameter("cDiscount");  
    double gst = calculateRate(covType, cDis, pricel) * 0.06;  
    double total = gst + calculateRate(covType, cDis, pricel);  
<%>
```


Lab Exercise

Compiled Successfully.

Output.

BMI Calculator

Weight :	<input type="text" value="65"/>	KILOGRAM
Height :	<input type="text" value="1.7"/>	METER
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>		

Body Mass Index (BMI) Calculator

Your Weight : 65.0 kg

Your Height : 1.7 m

Your BMI : 22.49

Weight Status : Optimal