UNIVERSITI MALAYSIA TERENGGANU

CSM3023 WEB BASED APPLICATION DEVELOPMENT

BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONORS

LAB 5

SEMESTER II 2023/2024

Prepared for:

SIR MOHD ARIZAL SHAMSIL BIN MAT RIFIN

Prepared by:

MUHAMMAD ZAHIER BIN RAZMI

(S67943)

Task 1: Using Scriplet to Access a Simple JavaBeans

Code:

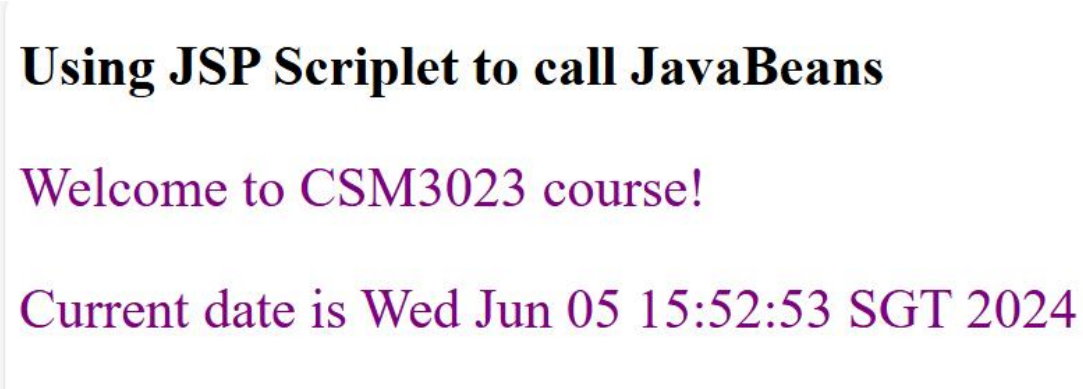
Message.java

```java
public class Message {

    private String msg;

    public Message() {
    }

    public Message(String msg) {
        this.msg = msg;
    }

    public String getMsg() {
        return msg;
    }

    public void setMsg(String msg) {
        this.msg = msg;
    }

}
```

Message1.jsp

```jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page info="Using JSP Standard Action to call JavaBeans" %>
<%@page import="java.util.Date, com.mycompany.lab5.Message" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Using JSP Scriplet</title>
        <style>
            p {
                font-size: 2rem;
                color: purple;
            }
        </style>
    </head>
    <body>
        <h1>Using JSP Scriplet to call JavaBeans</h1>

        <%
            Message objMsg = new Message();

            objMsg.setMsg("Welcome to CSM3023 course!");

            out.println("<p>" + objMsg.getMsg() + "</p>");

            out.println("<p>Current date is " + new Date() + "</p>");
        %>
    </body>
</html>
```

Output:



Reflection:

1. What you have learnt from this exercise?

   From this exercise, I have learned how to use JavaBeans in JSP (JavaServer Pages) and how to call them using JSP scriplets.

2. Explain the differences when calling JavaBeans using JSP Standard Action and Java Scriptlet.

   JSTL (JavaServer Pages Standard Tag Library) provides a set of tags that encapsulate core functionality common to many JSP applications. When using JSTL, JavaBeans can be included using tags like <jsp:useBean> to instantiate and manage beans, and <jsp:setProperty> to set properties on those beans. This approach promotes separation of concerns by keeping Java code out of the JSP file, making the code easier to read and maintain.

   JSP scriplets allow to directly embed Java code within JSP page using <% %> tag. While convenient for quick prototyping and small projects, scriplets can lead to code that's harder to maintain and debug, as it mixes presentation logic with business logic. Scriplets can make the JSP page cluttered and less readable, especially as the page grows in complexity.

Task 2: Problem Solving using JavaBeans

Code:

Register.java

```
public class Register {

    private String icNo, name;
    private int trainingType, paxNo, isStudent;

    public Register() {
    }

    public Register(String icNo, String name, int trainingType, int paxNo, int isStudent) {
        this.icNo = icNo;
        this.name = name;
        this.trainingType = trainingType;
        this.paxNo = paxNo;
        this.isStudent = isStudent;
    }

    public String getIcNo() {
        return icNo;
    }

    public void setIcNo(String icNo) {
        this.icNo = icNo;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
    public int getTrainingType() {
        return trainingType;
    }

    public void setTrainingType(int trainingType) {
        this.trainingType = trainingType;
    }

    public int getPaxNo() {
        return paxNo;
    }

    public void setPaxNo(int paxNo) {
        this.paxNo = paxNo;
    }

    public int getIsStudent() {
        return isStudent;
    }

    public void setIsStudent(int isStudent) {
        this.isStudent = isStudent;
    }

    public String getTrainingTypeStr() {
        switch (getTrainingType()) {
            case 1:
                return "C++ Training";
```

```
            case 2:
                return "Java for Beginner";

            case 3:
                return "HTML5";

            case 4:
                return "Java EEE";

            case 5:
                return "Android Programming";

            default:
                return "";
        }
    }

    public String getIsStudentStr() {
        if (getIsStudent() == 1) {
            return "Yes";
        }
        else {
            return "No";
        }
    }

    public double getAmount() {
        double amount;
        switch (getTrainingType()) {
            case 1:
                amount = 3000 * getPaxNo();
                break;

            case 2:
                amount = 3000 * getPaxNo();
                break;

            case 3:
                amount = 2800 * getPaxNo();
                break;

            case 4:
                amount = 5500 * getPaxNo();
                break;

            case 5:
                amount = 3200 * getPaxNo();
                break;

            default:
                amount = 0;
        }

        if (getIsStudent() == 1) {
            amount *= 0.9;
        }
        return amount;
    }
}
```

registerTraining.jsp

```jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Training Registration</title>
    </head>
    <body>
        <h1>Register IT Training</h1>

        <form action="processTraining.jsp">
            <fieldset>
                <legend><b>Training Registration</b></legend>
                <table>
                    <tbody>
                        <tr>
                            <td><label for="icNo">IC No</label></td>
                            <td><input type="text" name="icNo" id="icNo" placeholder="E.g. 911210-05-1234"></td>
                        </tr>
                        <tr>
                            <td><label for="name">Name</label></td>
                            <td><input type="text" name="name" id="name" placeholder="Enter your name"></td>
                        </tr>
                        <tr>
                            <td><label for="trainingType">Type of Training</label></td>
                            <td>
                                <select name="trainingType" id="trainingType">
                                    <option value="1">C++ Training</option>
                                    <option value="2">Java for Beginner</option>
                                    <option value="3">HTML5</option>
                                    <option value="4">Java EEE</option>
                                    <option value="5">Android Programming</option>
                                </select>
                            </td>
                        </tr>
                        <tr>
                            <td><label for="paxNo">No of Pax</label></td>
                            <td><input type="number" name="paxNo" id="paxNo" placeholder="No of pax" min="0"></td>
                        </tr>
                        <tr>
                            <td><label>Student</label></td>
                            <td>
                                <input type="radio" name="isStudent" id="yes" value="1">
                                <label for="yes">Yes</label>
                                <input type="radio" name="isStudent" id="no" value="0" checked="checked">
                                <label for="no">No</label>
                            </td>
                        </tr>
                        <tr>
                            <td>
                                <br>
                                <button type="submit">Submit</button>
                                <button type="reset">Cancel</button>
                            </td>
                        </tr>
                    </tbody>
                </table>
            </fieldset>
        </form>
    </body>
</html>
```

processTraining.jsp

```jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="com.mycompany.lab5.Register" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Training Registration</title>
        <style>
            p {
                font-size: 2rem;
                color: purple;
            }
        </style>
    </head>
    <body>
        <h1>Training Registration Acknowledgement</h1>

        <%
            String icNo, name;
            int trainingType, paxNo, isStudent;

            String trainingTypeStr, isStudentStr;
            int fees;

            icNo = request.getParameter("icNo");
            name = request.getParameter("name");
            trainingType = Integer.parseInt(request.getParameter("trainingType"));
            paxNo = Integer.parseInt(request.getParameter("paxNo"));
            isStudent = Integer.parseInt(request.getParameter("isStudent"));
           Register register = new Register(icNo, name, trainingType, paxNo, isStudent);
        %>

        <p>IC No : <%= register.getIcNo() %></p>
        <p>Name : <%= register.getName() %></p>
        <p>Type of Training : <%= register.getTrainingTypeStr() %></p>
        <p>Number of Pax : <%= register.getPaxNo() %> person/s</p>
        <p>Student : <%= register.getIsStudentStr() %></p>
        <p>Amount Due : RM <%= String.format("%.2f", register.getAmount()) %></p>
    </body>
</html>
```

Output:

Training Registration Acknowledgement

IC No : 980825-12-5403

Name : Zahier

Type of Training : Java EEE

Number of Pax : 1 person/s

Student : Yes

Amount Due : RM 4950.00

Reflection:

1. What have you learned from this exercise?

   From this exercise, I have learned how to create a JavaBean named Register to handle registration details for IT training.

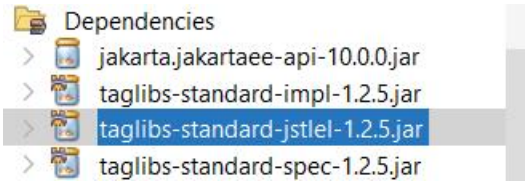
2. Describe the steps how you construct Register JavaBeans?

   - Determine the properties needed for registration, such as IC number, name, training type, number of participants, and student status. Declare private variables for each property in the Java class.
   - Implement a no-argument constructor and another constructor that accepts parameters for initializing the properties.
   - Generate getter and setter methods for each property to provide access to and modification of the class's private variables.

- Implement additional methods as needed for functionality like converting integer values to descriptive strings (getTrainingTypeStr() and getIsStudentStr()), and calculating registration fees (getAmount()).

Task 3 : Installing JSTL Taglibs

Output:



Reflection:

1. What you have learnt from this exercise?

   From this exercise I have learned how to install JSTL Taglibs in a web application project.

Task 4 : Using Java Standard Tag Library (JSTL)

Code Task 1:

jstlCore1.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Using JSTL Tag Library</title>
    </head>
    <body>
        <h1>Use JSTL's Features</h1>
        <c:set var="message" value="Welcome to CSM3023 - Web Programming courses..!" />
        <p><c:out value="${message}" /></p>
    </body>
</html>
```

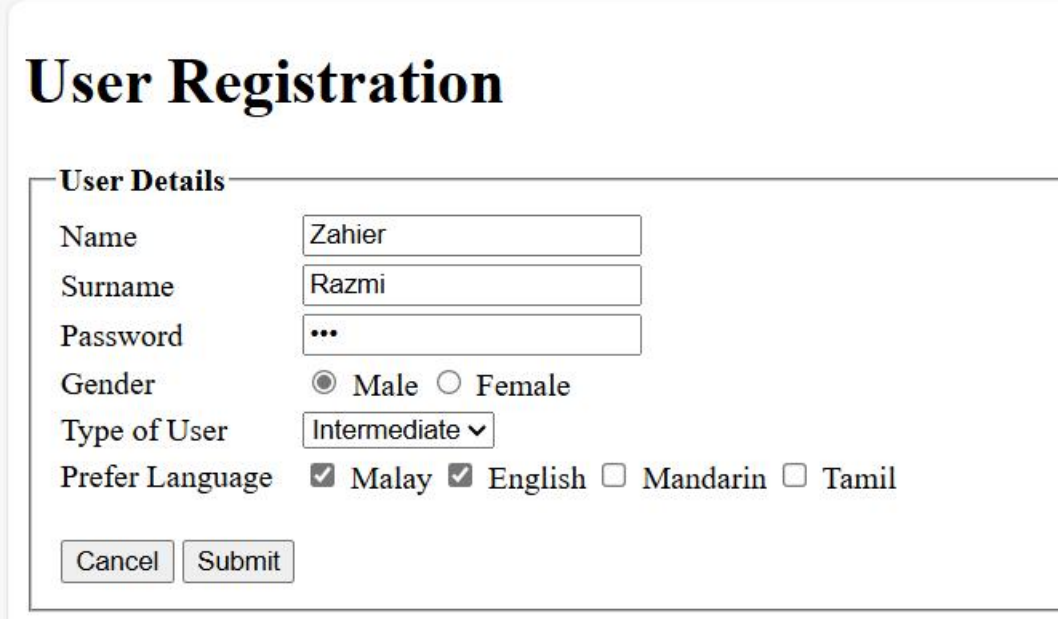
Output:

Code Task 2:

```
<!DOCTYPE html>
<html>
    <head>
        <title>User Registration</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <h1>User Registration</h1>

        <form action="processUser.jsp">
            <fieldset>
                <legend><b>User Details</b></legend>
                <table>
                    <tbody>
                        <tr>
                            <td><label for="userName">Name</label></td>
                            <td><input type="text" name="userName" id="userName" placeholder="Enter name"></td>
                        </tr>
                        <tr>
                            <td><label for="userSurname">Surname</label></td>
                            <td><input type="text" name="userSurname" id="userSurname" placeholder="Enter surname"></td>
                        </tr>
                        <tr>
                            <td><label for="userPassword">Password</label></td>
                            <td><input type="password" name="userPassword" id="userPassword" placeholder="Max 10 characters" maxlength="10"></td>
                        </tr>
```

```
                <tr>
                    <td>Gender</td>
                    <td>
                        <input type="radio" name="userGender" id="male" value="1">
                        <label for="male">Male</label>
                        <input type="radio" name="userGender" id="female" value="2">
                        <label for="female">Female</label>
                    </td>
                </tr>
                <tr>
                    <td>Type of User</td>
                    <td>
                        <select name="userType" id="userType">
                            <option value="1">Beginner</option>
                            <option value="2">Intermediate</option>
                            <option value="3">Advanced</option>
                        </select>
                    </td>
                </tr>
                <tr>
                    <td>Prefer Language</td>
                    <td>
                        <input type="checkbox" name="userLanguage" id="malay" value="1">
                        <label for="malay">Malay</label>
                        <input type="checkbox" name="userLanguage" id="english" value="2">
                        <label for="english">English</label>
                        <input type="checkbox" name="userLanguage" id="mandarin" value="3">
                        <label for="mandarin">Mandarin</label>
                         <input type="checkbox" name="userLanguage" id="tamil" value="4">
                         <label for="tamil">Tamil</label>
                    </td>
                </tr>
                <tr>
                    <td>
                        <br>
                        <button type="reset">Cancel</button>
                        <button type="submit">Submit</button>
                    </td>
                </tr>
            </tbody>
        </table>
    </fieldset>
</form>
</body>
</html>
```

Output Task 2:



Retrieve info using c:param & display it using c:out

Name: Zahier

Surname: Razmi

Gender: Male

Type of user: Intermediate

Prefer Language(s): Malay English

Code Task 3:

jstlFormat1.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSTL Formatting Tag</title>
    </head>
    <body>
        <h1>Using JSTL Formatting Tag for Formatting</h1>

        <c:set var="total" value="2880.4638" />
        <p>Number to be formatted is <c:out value="${total}" /></p>
        <p>Formatting number as currency with currency code: <fmt:formatNumber type="currency" currencyCode="MYR" value="${total}" /></p>
        <p>Formatting number to the nearest 2 integer digit: <fmt:formatNumber type="number" maxIntegerDigits="2" value="${total}" /></p>
        <p>Formatting number by grouping: <fmt:formatNumber type="number" groupingUsed="true" value="${total}" /></p>
        <p>Formatting number to the 3 decimal places: <fmt:formatNumber type="number" groupingUsed="false" maxFractionDigits="3" value="${total}" /
        <p>Formatting number to percentage: <fmt:formatNumber type="percent" groupingUsed="false" value="${total}" /></p>
    </body>
</html>
```

Output Task 3:

Using JSTL Formatting Tag for Formatting

Number to be formatted is 2880.4638

Formatting number as currency with currency code: MYR2,880.46

Formatting number to the nearest 2 integer digit: 80.464

Formatting number by grouping: 2,880.464

Formatting number to the 3 decimal places: 2880.464

Formatting number to percentage: 288046%

Reflection:

1. What the purpose of using JSTL’s tag library?

   The purpose of using JSTL's tag library is to provide a standardized set of tags that facilitate common tasks in JavaServer Pages (JSP) development. These tags encapsulate commonly needed functionality, reducing the need for embedded Java code (scriplets) within JSP pages. This promotes cleaner, more maintainable code by separating presentation logic from business logic.

2. List FIVE(5) categories of JSTL library.

   - Core
   - Formatting
   - XML
   - SQL
   - Functions

Task 5 : Using JSP Standard Tag Library

Code Task 1:

fmt_formatDate.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>fmt:formatDate Example</title>
    </head>
    <body>
        <h2>fmt:formatDate Examples</h2>

        <c:set var="now" value="<%= new java.util.Date() %>" />
        <p>
            Time (fmt:formatDate type="time"):
            <strong>
                <fmt:formatDate type="time" value="${now}" />
            </strong>
        </p>
        <p>
            Date (fmt:formatDate type="date"):
            <strong>
                <fmt:formatDate type="date" value="${now}" />
            </strong>
        </p>
        <p>
            Date, Time (fmt:formatDate type="both"):
            <strong>
                <fmt:formatDate type="both" value="${now}" />
            </strong>
        </p>
        <p>
            Date, Time Short (fmt:formatDate type="both" dateStyle="short" timeStyle="short"):
            <strong>
                <fmt:formatDate type="both" dateStyle="short" timeStyle="short" value="${now}" />
            </strong>
        </p>
        <p>
            Date, Time Medium (fmt:formatDate type="both" dateStyle="medium" timeStyle="medium"):
            <strong>
                <fmt:formatDate type="both" dateStyle="medium" timeStyle="medium" value="${now}" />
            </strong>
        </p>
        <p>
            Date, Time Long (fmt:formatDate type="both" dateStyle="long" timeStyle="long"):
            <strong>
                <fmt:formatDate type="both" dateStyle="long" timeStyle="long" value="${now}" />
            </strong>
        </p>
        <p>
            Date, Time (dd-MM-yyyy HH:mm:ss):
            <strong>
                <fmt:formatDate pattern="dd-MM-yyyy HH:mm:ss" value="${now}" />
            </strong>
        </p>
        <fmt:formatDate pattern="dd-MM-yyyy HH:mm" value="${now}" var="nowString" />
        <p>
            Now String (dd-MM-yyyy HH:mm):
            <strong>
                <c:out value="${nowString}" />
            </strong>
        </p>
    </body>
</html>
```

Output Task 1:

fmt:formatDate Examples

Time (fmt:formatDate type="time"): **10:34:59 PM**

Date (fmt:formatDate type="date"): **Jun 5, 2024**

Date, Time (fmt:formatDate type="both"): **Jun 5, 2024, 10:34:59 PM**

Date, Time Short (fmt:formatDate type="both" dateStyle="short" timeStyle="short"): **6/5/24, 10:34 PM**

Date, Time Medium (fmt:formatDate type="both" dateStyle="medium" timeStyle="medium"): **Jun 5, 2024, 10:34:59 PM**

Date, Time Long (fmt:formatDate type="both" dateStyle="long" timeStyle="long"): **June 5, 2024, 10:34:59 PM SGT**

Date, Time (dd-MM-yyyy HH:mm:ss): **05-06-2024 22:34:59**

Now String (dd-MM-yyyy HH:mm): **05-06-2024 22:34**

Code Task 2:

fmt_parseDate.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>fmt:parseDate Examples</title>
    </head>
    <body>
        <h2>fmt:parseDate Examples</h2>

        <c:set var="dateTimeString" value="17-11-2015 11:49" />
        <h4>
            dateTimeString:
            <c:out value="${dateTimeString}" />
        </h4>

        <!-- Parsing a date time string, and store in a variable type of java -->
        <fmt:parseDate value="${dateTimeString}" type="both" var="parsedDateTime" pattern="dd-MM-yyyy HH:mm" />
        <p>
            The date after parsing:
            <c:out value="${parsedDateTime}" />
        </p>
        <br>
        <p>
            Date only (dd/MM/yyyy):
            <fmt:formatDate value="${parsedDateTime}" pattern="dd/MM/yyyy" />
        </p>
    </body>
</html>
```

Output Task 2:

Reflection:

1. What you have learnt from this exercise?

   From this exercise I have learned how to format dates, parse dates, and combine formatting with core tags.

Exercise 1

Code:

circle.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Find area of a circle</title>
    </head>
    <body>
        <h1>Find area of a circle</h1>

        <form>
            <table>
                <tr>
                    <td><label>Radius</label></td>
                    <td><input type="number" name="radius" required></td>
                    <td><button type="submit">Calculate</button></td>
                </tr>
            </table>
        </form>

        <c:if test="${param.radius != null}" >
            <c:set var="radius" value="${param.radius}" />
            <c:set var="area" value="${Math.PI * radius * radius}" />
            <c:set var="perimeter" value="${2 * Math.PI * radius}" />

            <p>Radius: <c:out value="${radius}" /></p>
            <p>Area: <fmt:formatNumber type="number" groupingUsed="false" maxFractionDigits="3" value="${area}" /></p>
            <p>Perimeter: <fmt:formatNumber type="number" groupingUsed="false" maxFractionDigits="3" value="${perimeter}" /></p>
        </c:if>
    </body>
</html>
```

Output:

Exercise 2

Code:

StockPurchase.java

```
public class StockPurchase {

    private int numShares;
    private double pricePerShare;
    private double commissionRate;

    public StockPurchase() {
    }

    public StockPurchase(int numShares, double pricePerShare, double commissionRate) {
        this.numShares = numShares;
        this.pricePerShare = pricePerShare;
        this.commissionRate = commissionRate;
    }

    public int getNumShares() {
        return numShares;
    }

    public void setNumShares(int numShares) {
        this.numShares = numShares;
    }

    public double getPricePerShare() {
        return pricePerShare;
    }

    public void setPricePerShare(double pricePerShare) {
        this.pricePerShare = pricePerShare;
    }

    public double getCommissionRate() {
        return commissionRate;
    }

    public void setCommissionRate(double commissionRate) {
        this.commissionRate = commissionRate;
    }

    public double getCommissionRateAsPercentage() {
        if (commissionRate <= 1)
            return commissionRate * 100;
        else
            return commissionRate;
    }

    public double calculateStockCost() {
        return getNumShares() * getPricePerShare();
    }

    public double calculateCommission() {
      return calculateStockCost() * getCommissionRate();
    }

     public double calculateTotalCost() {
       return calculateStockCost() + calculateCommission();
     }

}
```

Stock_purchase.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Stock Purchase</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <h1>Stock Purchase</h1>

        <form action="stock_purchase_process.jsp">
            <table>
                <tr>
                    <td><label for="shareNum">Number of Shares:</label></td>
                    <td><input type="number" name="shareNum" id="shareNum" min="0" required></td>
                </tr>
                <tr>
                    <td><label for="sharePrice">Price per Share:</label></td>
                    <td><input type="number" name="sharePrice" id="sharePrice" min="0" step="0.01" required></td>
                </tr>
                <tr>
                    <td><label for="comRate">Commission Rate (%):</label></td>
                    <td><input type="number" name="comRate" id="comRate" min="0" max="100" required></td>
                </tr>
                <tr>
                    <td>
                        <br>
                        <button type="submit">Calculate</button>
                    </td>
                </tr>
            </table>
        </form>
    </body>
</html>
```

Stock_purchase_process.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Stock Purchase Results</title>
    </head>
    <body>
        <c:set var="shareNum" value="${param.shareNum}" />
        <c:set var="sharePrice" value="${param.sharePrice}" />
        <c:set var="comRate" value="${param.comRate}" />

        <jsp:useBean id="purchase" class="com.mycompany.lab5.StockPurchase">
            <jsp:setProperty name="purchase" property="numShares" value="${shareNum}" />
            <jsp:setProperty name="purchase" property="pricePerShare" value="${sharePrice}" />
            <jsp:setProperty name="purchase" property="commissionRate" value="${comRate / 100}" />
        </jsp:useBean>

        <h2>Stock Purchase Details</h2>
        <p>Number of Share: <c:out value="${purchase.getNumShares()}" /></p>
        <p>Price per Share: <fmt:formatNumber type="currency" currencySymbol="RM" groupingUsed="false" maxFractionDigits="2" value="${purchase.getP
        <p>Commission Rate: <fmt:formatNumber type="number" groupingUsed="false" maxFractionDigits="0" value="${purchase.getCommissionRateAsPercent
        <br>

        <h2>Stock Purchase Results</h2>
        <p>Stock Cost: <fmt:formatNumber type="currency" currencySymbol="RM" groupingUsed="false" maxFractionDigits="2" value="${purchase.calculate
        <p>Commission: <fmt:formatNumber type="currency" currencySymbol="RM" groupingUsed="false" maxFractionDigits="2" value="${purchase.calculate
        <p>Total Cost: <fmt:formatNumber type="currency" currencySymbol="RM" groupingUsed="false" maxFractionDigits="2" value="${purchase.calculate
    </body>
</html>
```

Output:

Stock Purchase

Number of Shares: 800

Price per Share: 10.5

Commission Rate (%): 5

Calculate

Stock Purchase Details

Number of Share: 800

Price per Share: RM10.50

Commission Rate: 5%

Stock Purchase Results

Stock Cost: RM8400.00

Commission: RM420.00

Total Cost: RM8820.00