

## Task 1: Using JSP Page to Access a Simple MySQL Database

**Coding:**

### SampleInsertionRecord.jsp

```
<%--
    Document   : SampleInsertionRecord
    Created on : 18 May 2024, 00:48:02
    Author    : Atirah s67397
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page language="java" %>
<%@page import="java.sql.*" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Task 1</title>
    </head>
    <body>
        <h1>Lab 6 - Task 1 - Sample Insertion records into MySQL
through</h1>
        <%
            int result;

            //step 1 : load JDBC driver..
            Class.forName("com.mysql.jdbc.Driver");
            out.println("Step 1 : MYSQL driver loaded..!");
        %>
        <br>
        <%
            //step 2 : establish the connection...
```

```

String myURL = "jdbc:mysql://localhost/csm3023";
Connection myConnection =
DriverManager.getConnection(myURL,"root","admin");
    out.println("Step 2 : Database is connected..!");
%>
<br>
<%
    //step 3 : create a PreparedStatement object...
    out.println("Step 3 : Prepared Statement created..!");

    //prepared SQL Query as a string..
    String sInsertQry = "INSERT INTO FirstTable VALUE(?)";

    // Call method prepareStatement
    PreparedStatement myPS =
myConnection.prepareStatement(sInsertQry);
%>
<br>
<%
    //assign each value to respective columns for Book's table..(C-Create)
    out.println("Step 4 : Perform insertion of record..!");
    String name = "Welcome to access MYSQL database with JSP...!";
    myPS.setString(1,name);

    result = myPS.executeUpdate();

    if(result>0){
%>
<br>
<%
        out.println("Step 5 : Close databse connection..!");
        out.println(" ");
        out.println("Database connection is closed..!");

```

```

        out.print("<p>" + "The record : (" + name
            + ") is successfully created..!" + "</p>");
    }
    //step 5 : close database connection..!
    myConnection.close();
%>
<br>
</body>
</html>

```

## MYSQL:



```

1 • create table FirstTable(
2     NAME Varchar(45)
3 );
4
5 • select * from FirstTable;

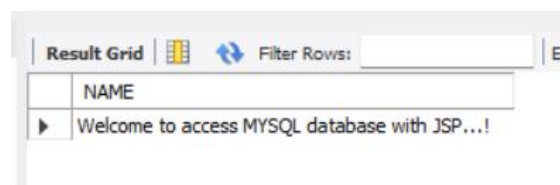
```

## Output:

### Lab 6 - Task 1 - Sample Insertion records into MySQL through

Step 1 : MYSQL driver loaded..!  
 Step 2 : Database is connected..!  
 Step 3 : Prepared Statement created..!  
 Step 4 : Perform insertion of record..!  
 Step 5 : Close database connection..! Database connection is closed..!

The record : (Welcome to access MYSQL database with JSP...!) is successfully created..!



NAME
Welcome to access MYSQL database with JSP...!

## Task 2: Create Records via JSP Page

## Coding:

### insertAuthor.jsp

```
<%--
    Document : insertAuthor
    Created on : 18 May 2024, 01:21:02
    Author : Nur Atirah (S67397)
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Task 2</title>
        <style>
            label {
                display: inline-block;
                width: 150px;
                margin-right: 10px;
            }
        </style>
    </head>
    <body>
        <h1>Lab 6 - Task 2 - Perform creating and retrieving records via JSP
page</h1>
        <form action="processAuthor.jsp" method="post">
            <fieldset>
                <legend>Author Registration</legend>
                <label for="authno">Author No</label> <input type="text"
id="authno" name="authno" placeholder="E.g.: UKXXXXXX">
                <br><br>
                <label for="name">Name</label> <input type="text" id="name">
            </fieldset>
        </form>
    </body>
</html>
```

```

name="name" placeholder="Enter your name">
    <br><br>
    <label    for="address">Address</label>    <input    type="text"
id="address" name="address" placeholder="Enter you address">
    <br><br>
    <label    for="city">City</label>    <input    type="text"    id="city"
name="city" placeholder="Enter your city">
    <br><br>
    <label    for="state">State</label>    <input    type="text"    id="state"
name="state" placeholder="Enter your state">
    <br><br>
    <label    for="zip">Zip</label>    <input    type="number"    id="zip"
name="zip" placeholder="Enter your zip">
    <br><br>

    <input    type="submit"    value="Submit"    >    <input    type="reset"
value="Cancel">
    </fieldset>
</form>
</body>
</html>

```

## Author.java

```

package lab6.com;

/*
 *      Click    nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
 */

```

```
/**
 *
 * nur atirah s67397
 */
public class Author {
    private String authno;
    private String name;
    private String address;
    private String city;
    private String state;
    private String zip;

    /**
     * @return the authno
     */
    public String getAuthno() {
        return authno;
    }

    /**
     * @param authno the authno to set
     */
    public void setAuthno(String authno) {
        this.authno = authno;
    }

    /**
     * @return the name
     */
    public String getName() {
        return name;
    }
}
```

```
/**
 * @param name the name to set
 */
public void setName(String name) {
    this.name = name;
}

/**
 * @return the address
 */
public String getAddress() {
    return address;
}

/**
 * @param address the address to set
 */
public void setAddress(String address) {
    this.address = address;
}

/**
 * @return the city
 */
public String getCity() {
    return city;
}

/**
 * @param city the city to set
 */
public void setCity(String city) {
```

```
        this.city = city;
    }

    /**
     * @return the state
     */
    public String getState() {
        return state;
    }

    /**
     * @param state the state to set
     */
    public void setState(String state) {
        this.state = state;
    }

    /**
     * @return the zip
     */
    public String getZip() {
        return zip;
    }

    /**
     * @param zip the zip to set
     */
    public void setZip(String zip) {
        this.zip = zip;
    }
}
```



## processAuthor.jsp

```
<%--
    Document : processAuthor
    Created on : 18 May 2024, 03:45:01
    Author : Nur Atirah
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page language="java" %>
<%@page import="java.sql.*" %>
<jsp:useBean id="myAuthor" class="lab6.com.Author" scope="request"/>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Task 2</title>
    </head>
    <body>
        <h1>Lab 6 - Task 2 - Perform creating and retrieving records via JSP
page</h1>
        <jsp:setProperty name="myAuthor" property="*" />

        <%
            int result;

            Class.forName("com.mysql.jdbc.Driver");

            String myURL = "jdbc:mysql://localhost/csm3023";
            Connection myConnection =
DriverManager.getConnection(myURL,"root","admin");

            String sInsertQry = "INSERT INTO Author (authno, name, address,
city, state, zip) VALUES (?, ?, ?, ?, ?, ?)";
```

PreparedStatement	myPS	=
-------------------	------	---

```

myConnection.prepareStatement(sInsertQry);

myPS.setString(1, myAuthor.getAuthno());
myPS.setString(2, myAuthor.getName());
myPS.setString(3, myAuthor.getAddress());
myPS.setString(4, myAuthor.getCity());
myPS.setString(5, myAuthor.getState());
myPS.setString(6, myAuthor.getZip());

result = myPS.executeUpdate();

if (result > 0){
    out.println("\tRecord successfully added into Author table..!");
    out.print("<p>" + "Record with author no " + myAuthor.getAuthno()
    + " successfully created..!" + "</p>");
    out.print("<p>" + "Details of record are: " + "</p>");
    out.print("<p>Name " + myAuthor.getName() + "</p>");
    out.print("<p>Address " + myAuthor.getAddress() + "</p>");
    out.print("<p>City " + myAuthor.getCity() + "</p>");
    out.print("<p>State " + myAuthor.getState() + "</p>");
    out.print("<p>Zip " + myAuthor.getZip() + "</p>");
}

//step 5: close database connection..!
System.out.println("Step 5 : Close database connection..!");
myConnection.close();
System.out.println(" ");
System.out.println("Database connection is closed..!");
%>
</body>
</html>

```

## MYSQL:

```
7 • create table Author(  
8     authno varchar(15) primary key,  
9     name varchar(40),  
10    address varchar(40),  
11    city varchar(40),  
12    state varchar(40),  
13    zip varchar(40)  
14 );
```

## Output:

### Lab 6 - Task 2 - Perform creating and retrieving records via JSP page

Author Registration	
Author No	<input type="text" value="E.g.: UKXXXXXX"/>
Name	<input type="text" value="Enter your name"/>
Address	<input type="text" value="Enter you address"/>
City	<input type="text" value="Enter your city"/>
State	<input type="text" value="Enter your state"/>
Zip	<input type="text" value="Enter your zip"/>
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

### Lab 6 - Task 2 - Perform creating and retrieving records via JSP page

Record successfully added into Author table..!

Record with author no s67397 successfully created..!

Details of record are:

Name NUR ATIRAH BINTI SHAMSUDDIN

Address Bandar Cheneh Baru

City Chukai

State Terengganu

Zip 24000

	authno	name	address	city	state	zip
▶	s67397	NUR ATI...	Bandar Cheneh Baru	Chukai	Terengganu	24000
*	NULL	NULL	NULL	NULL	NULL	NULL

## Reflection

1. What have you learnt from this exercise?

This exercise taught me how to effectively use JSP and JavaBeans to build a web application that interacts with a MySQL database, focusing on data input, processing, and storage.

2. Define step by step before you successfully perform the transaction in a database.

Before successfully executing a database transaction, you need to create a database table with required fields, design a JSP form to capture user input, develop a JavaBean to encapsulate the form data, establish a JDBC connection in the JSP file, execute an SQL insert statement to store data, handle exceptions and display transaction results appropriately, and finally, close the database connection to ensure proper resource management.

### Task 3: Create Records Constrained by Regular Expression

**Coding:**

**insertStudent:**

```
<%--
  Document   : insertStudent
  Created on : 18 May 2024, 19:21:41
  Author      : nur atirah s67397
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Task 3</title>
<style>
  label {
    display: inline-block;
    width: 150px;
    margin-right: 10px;
  }
</style>
</head>
<body>
  <h1>Inserting Student Information!</h1>
  <form action="processStudent.jsp" method="post">
    <fieldset>
      <legend>Student Registration</legend>
      <label>Student      No</label><input      type="text"      id="stuno"
name="stuno" placeholder="E.g.: UKXXXXXX" required>
      <br><br>
      <label>Name</label><input type="text" id="name" name="name"
placeholder="Enter your name" required>
      <br><br>
      <label for="program">Program</label>
      <select id="program" name="program" required>
        <option value="BSc. Soft. Eng.">BSc. Soft. Eng.</option>
        <option value="BSc. with IM">BSc. with IM</option>
        <option value="BSc. in Networking">BSc. in Networking</option>
        <option value="BSc. in Robotics">BSc. in Robotics</option>
      </select>
      <br><br>
      <input      type="submit"      value="Submit">      <input      type="reset"
value="Cancel">
    </fieldset>
  </form>
```

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

## Student JavaBeans

```
/*
 *      Click      nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
 */
package lab6.com;

import java.util.regex.Matcher;
import java.util.regex.Pattern;

/**
 *
 * @author Atirah s67397
 */

public class Student {

    private String stuno;
    private String name;
    private String program;

    public String getStuno() {
        Pattern pt = Pattern.compile("[A-Za-z0-9]*");
        Matcher mt = pt.matcher(stuno);
        boolean b1 = mt.matches();
        String valid;
        if (b1) {
```

```
        valid = stuno;
    } else {
        valid = "invalid";
    }
    return valid;
}

public void setStuno(String stuno) {
    this.stuno = stuno;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public String getProgram() {
    return program;
}

public void setProgram(String program) {
    this.program = program;
}
}
```

**processStudent.jsp**

```

<%--
    Document : processBook
    Created on : 19 May 2024, 00:05:00
    Author : nur atirah s67397
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page language="java" %>
<%@page import="java.sql.*" %>
<%@page errorPage="errorStudent.jsp" %>
<jsp:useBean id="myStudent" class="lab6.com.Student" scope="request" />
<jsp:setProperty name="myStudent" property="*" />
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Task 3</title>
    </head>
    <body>
        <%
            String studentNo = myStudent.getStuno();
            if ("invalid".equals(studentNo)) {
                out.println("<p>Invalid student number format. Please try
again.</p>");
            } else {
                try {
                    // Step 1: Load JDBC driver
                    Class.forName("com.mysql.cj.jdbc.Driver");
                    System.out.println("Step 1 : MYSQL driver loaded..!");

                    // Step 2: Establish the connection
                    String myURL = "jdbc:mysql://localhost/csm3023";
                    Connection myConnection =
DriverManager.getConnection(myURL, "root", "admin");

```



```

        System.out.println("Step 2 : Database is connected.!");

        // Step 3: Create a PreparedStatement object
        String sInsertQry = "INSERT INTO Student (stuno, stuname,
stuprogram) VALUES (?, ?, ?)";

        PreparedStatement myPS =
myConnection.prepareStatement(sInsertQry);

        // Step 4: Set the parameters
        myPS.setString(1, myStudent.getStuno());
        myPS.setString(2, myStudent.getName());
        myPS.setString(3, myStudent.getProgram());

        // Step 5: Execute the query
        int result = myPS.executeUpdate();
        if (result > 0) {
            out.println("<p>Record with student no " +
myStudent.getStuno() + " successfully created..!</p>");
            out.println("<p>Details of record are:</p>");
            out.println("<p>Student ID: " + myStudent.getStuno() + "</p>");
            out.println("<p>Name: " + myStudent.getName() + "</p>");
            out.println("<p>Program: " + myStudent.getProgram() +
"</p>");
        }

        // Step 6: Close the connection
        myConnection.close();
    } catch (Exception e) {
        e.printStackTrace();
        out.println("<p>Error: " + e.getMessage() + "</p>");
    }
}
%>

```

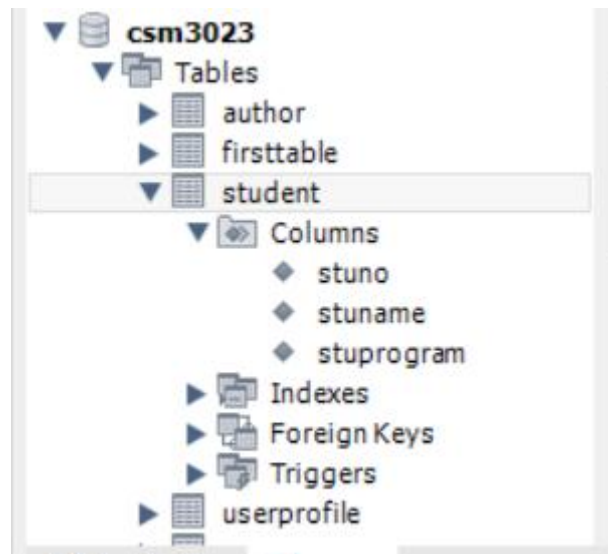
```
</body>
</html>
```

### **errorStudent.jsp**

```
<%--
  Document   : errorStudent
  Created on : 19 May 2024, 00:22:39
  Author      : nur atirah s67397
--%>

<%@page      contentType="text/html"      pageEncoding="UTF-8"
isErrorPage="true"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Task 3</title>
  </head>
  <body>
    <form id="errorFrm" action="insertStudent.jsp" method="post">
      <h1>Lab 6 - Task 4 - Perform creating and retrieving records via JSP
page when inserting record...!</h1>
      <p><jsp:expression>exception.getMessage()</jsp:expression></p>
      <br>
    </form>
  </body>
</html>
```

### **MYSQL:**



**Output:**

## Inserting Student Information!

Student Registration

Student No

Name

Program

▼

Record with student no UK67359 successfully created..!

Details of record are:

Student ID: UK67359

Name: sam

Program: BSc. with IM

## Reflection

### 1. What have you learnt from this exercise?

This exercise taught me how to create a web form using JSP (insertStudent.jsp) to collect student information, utilize a JavaBean (Student.java) for data handling, implement server-side processing in JSP (processStudent.jsp) to insert data into a MySQL database (csm3023), and handle errors with a dedicated error page (errorStudent.jsp).

### 2. Define step by step before you successfully perform the transaction in a database.

**Step 1:** Create a JSP form (insertStudent.jsp) with fields for student number, name, and program selection.

**Step 2:** Develop a JavaBean (Student.java) with getters and setters to encapsulate form data.

**Step 3:** In processStudent.jsp, validate student number format using regex in the JavaBean.

**Step 4:** Load the MySQL driver, establish a database connection, and prepare a PreparedStatement.

**Step 5:** Set parameters from the JavaBean, execute the SQL INSERT statement, and handle successful transaction outcomes by displaying inserted data.

**Step 6:** Close the database connection and handle exceptions using errorStudent.jsp to redirect and display error messages.

## Task 4: Perform Retrieving Records Via JSP Page

### Coding:

#### queryStudent:

```
<%--
    Document : queryStudent
    Created on : 25 May 2024, 12:39:58
    Author : Atirah s67397
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import = " java.sql.*" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Lab 6 - Task 3</title>
        <style>
            table{
                border-collapse: collapse;

            }

            td, th{
                border: 1px solid #999;
                padding: 0.5rem;
                text-align: left;
            }
            th{
                background: gold;
            }
        </style>
    </head>
```

```

<body>
  <h1>Lab 6 - Task 4 : Retrieving record via JSP Page.</h1>
  <%
    out.print("<table>");
    out.print("<thead>");
    out.print("<tr>");
    out.print("<th>" + "ISBNNo" + "</th>");
    out.print("<th>" + "Author" + "</th>");
    out.print("<th>" + "Title" + "</th>");
    out.print("</tr>");
    out.print("</thead>");
    out.print("<tbody>");

    //step 1 : load JDBC driver..
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Step 1 : MYSQL driver loaded..!");

    //step 2 : establish the connection..
    String myURL = "jdbc:mysql://localhost/csm3023";
    Connection myConnection =
DriverManager.getConnection(myURL,"root","admin");
    System.out.println("Step 2 : Database is connected..!");

    //step 3 : create a PreparedStatement object...
    Statement myStatement = myConnection.createStatement();

    //step 4: perform retrieve record from Student's table ... (R-retrieve)
    String myQuery = "SELECT * FROM student";
    ResultSet myResultSet = myStatement.executeQuery(myQuery);

    while (myResultSet.next()){
      out.print("<tr>");
      out.print("<td width=\"20%\">" +myResultSet.getString(1) +

```

```

"</td>");

        out.print("<td width=\"40%\">" +myResultSet.getString(2) +
"</td>");

        out.print("<td width=\"40%\">" +myResultSet.getString(3) +
"</td>");

        out.print("</tr>");
    }

    //step 5 : close the database connnection ..!
    System.out.println("Step 5 : Close database connection..!");
    myConnection.close();
    System.out.println("");
    System.out.println("Database connection is closed..!");

    out.print("</tbody>");
    out.print("</table>");

    %>
</body>
</html>

```

**Output:**

### Lab 6 - Task 4 : Retrieving record via JSP Page.

ISBNNo	Author	Title
S67393	NUR ATIRAH BINTI SHAMSUDDIN	BSc. Soft. Eng.
t7854	sss	BSc. Soft. Eng.
UK12489	Ahmad Salam	BSc. with IM
UK56789	Rosnaj Azman	BSc. Soft. Eng.
UK67342	Liew Cheng Huat	BSc. in Robotics
UK67355	Liew Cheng sgdj	BSc. in Networking
UK67359	sam	BSc. with IM

**Reflection**

1. What have you learnt from this exercise?

I have learned how to create a JSP-based web application that inserts student records into a MySQL database and retrieves them. I gained practical experience with form handling, database connectivity using JDBC, error handling, and displaying data in a tabular format.

2. Explain the differences when using Statement() and PreparedStatement().

**Statement():**

- Executes SQL queries directly.
- Vulnerable to SQL injection attacks because it does not handle parameterization.
- Typically used for simple queries with static SQL.

**PreparedStatement():**

- Provides a mechanism for pre-compiling SQL queries, allowing for parameterized queries.
- Enhances security by preventing SQL injection attacks.
- More efficient for executing the same query multiple times with different parameters.

## **Task 5: Create A Record Using JSP Model 1**

**Coding:**

**registerMarathon**

```
<!DOCTYPE html>
<html>
  <head>
```



```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Task 5</title>
<style>
    label {
        display: inline-block;
        width: 150px;
        margin-right: 10px;
    }
</style>
</head>
<body>
    <h1>Marathon</h1>
    <form action="processMarathon.jsp" method="post">
        <fieldset>
            <legend>Marathon Registration</legend><br>
            <label>IC No</label><input type="text" id="icno" name="icno"
placeholder="E.g.: 921110-10-2514">
            <br><br>
            <label>Name</label><input type="text" id="name" name="name"
placeholder="Enter your name">
            <br><br>
            <label for="category">Category</label>
            <select id="category" name="category">
                <option value="5 KM">5 KM</option>
                <option value="7 KM">7 KM</option>
                <option value="10 KM">10 KM</option>
            </select>
            <br><br>
            <input type="submit" value="Submit">
            <input type="reset" value="Cancel">
        </fieldset>
    </form>
</body>
```

</html>

## Marathon JavaBeans

```
package lab6.com;

/*
 *      Click      nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
 */

/**
 *
 * @author atirah s67397
 */
public class Marathon {
    private String icno;
    private String name;
    private String category;

    /**
     * @return the icno
     */
    public String getIcno() {
        return icno;
    }

    /**
     * @param icno the icno to set
     */
    public void setIcno(String icno) {
```

```
        this.icno = icno;
    }

    /**
     * @return the name
     */
    public String getName() {
        return name;
    }

    /**
     * @param name the name to set
     */
    public void setName(String name) {
        this.name = name;
    }

    /**
     * @return the category
     */
    public String getCategory() {
        return category;
    }

    /**
     * @param category the category to set
     */
    public void setCategory(String category) {
        this.category = category;
    }
}
```

## Database

```
/*
 *      Click      nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
 */
package lab6.com;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import lab6.com.Marathon;

/**
 * Bean : Database.java
 * Author : Nur Atirah s67397
 * Date : 25/05/2024
 */

public class Database {
    private static Connection myConnection = null;
    private static String myURL = "jdbc:mysql://localhost/csm3023";
    private int result = 0;

    public static Connection getConnection () throws ClassNotFoundException{
```

```

        if (myConnection != null){
            return myConnection;
        }
        else try {
            Class.forName("com.mysql.jdbc.Driver");
            myConnection =
DriverManager.getConnection(myURL,"root","admin");
        }
        catch(SQLException e){
            e.printStackTrace();
        }
        return myConnection;
    }

    public void closeConnection() throws ClassNotFoundException{
        try{
            myConnection.close();
        }
        catch(SQLException e){
            e.printStackTrace();
        }
    }
}

```

### MarathonDAO.java

```

/*
 *      Click      nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template

```

```

*/
package lab6.com;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import lab6.com.Database;

/**
 * Bean : MarathonDAO.java
 * Author : Nur Atirah s67397
 * Date : 25/05/2024
 */

public class MarathonDAO {
    private Connection connection;
    private int result = 0;
    public MarathonDAO() throws ClassNotFoundException{
        connection = Database.getConnection();
    }
    public int addDetails (Marathon marathon){
        try{
            String mySQL = "INSERT INTO marathon(icno, name, category)
values (?, ?, ?)";
            PreparedStatement preparedStatement =
connection.prepareStatement(mySQL);

            System.out.println("IC NO      =" + marathon.getIcno());
            System.out.println("Name      =" + marathon.getName());
            System.out.println("Category  =" + marathon.getCategory());

            //parameters

```

```

        preparedStatement.setString(1, marathon.getIcno());
        preparedStatement.setString(2, marathon.getName());
        preparedStatement.setString(3, marathon.getCategory());
        result = preparedStatement.executeUpdate();

    }catch (SQLException e ){
        e.printStackTrace();
    }
    return result;
}
}

```

### **processMarathon.jsp**

```

<%--
    Document   : processMarathon
    Created on : 5 Jun 2024, 16:50:01
    Author    : atirah s67397
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.sql.*" %>
<%@page import="lab6.com.Database" %>
<%@page import="lab6.com.Marathon" %>
<%@page import="lab6.com.MarathonDAO" %>
<jsp:useBean id="myMarathon" class="lab6.com.Marathon" scope="request"
/>
<jsp:setProperty name="myMarathon" property="icno" param="icno" />
<jsp:setProperty name="myMarathon" property="name" param="name" />
<jsp:setProperty name="myMarathon" property="category" param="category"
/>

<!DOCTYPE html>

```

```

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <%
      int result;

      //step 1 create db obj
      Database myDB = new Database();

      MarathonDAO object1 = new MarathonDAO();

      //step2 add records
      result = object1.addDetails(myMarathon);

      //step3 determine whether the transaction is success..
      if (result > 0){
        out.print("<p>Record with IC No " + myMarathon.getIcno() + "
successfully created ..! </p>");
        out.print("<p>Ic No   : " + myMarathon.getIcno() + "</p>");
        out.print("<p>Name     : " + myMarathon.getName() + "</p>");
        out.print("<p>Category : " + myMarathon.getCategory() + "</p>");
      } else {
        out.print("<p>Failed to add record with IC No   " +
myMarathon.getIcno() + "</p>");
      }

      //step 4 close database connection..!
      myDB.closeConnection();
    %>
  </body>

```



</html>

## MYSqL:

```
17
18 • create table marathon(
19     icno varchar(14) primary key,
20     name varchar (100),
21     category varchar (54)
22 );
23
24 • select * from marathon;
```

## Output:

# Marathon

Marathon Registration

IC No

E.g.: 921110-10-2514

Name

Enter your name

Category

5 KM ▾

Submit

Cancel

Record with IC No 031130-06-0602 successfully created ..!

Ic No : 031130-06-0602

Name : NUR ATIRAH BINTI SHAMSUDDIN

Category : 5 KM

## Reflection

1. What have you learnt from this exercise?

I have learned how to handle forms in JSP, use JavaBeans to manage form data, connect to and interact with a database using JDBC, incorporate Java code within JSP for dynamic content, and debug issues related to data handling and output.

2. Describe the benefits of using JavaBeans.

JavaBeans encapsulate data, making it safe and organized. They are reusable across different parts of an application, which promotes code reuse. JavaBeans help separate business logic from presentation logic, simplifying data management and following standard design practices. They integrate easily with JSP, facilitating data handling, and make the code easier to update and maintain.

## Exercise

Implement user login

1. Create a table known as userprofile using database schema CF3107 using these attributes.

- username as a character length 15 and must be primary key
- password as a character length 10
- firstname as varchar(50)
- lastname as varchar(50)

2. Create insertUser.html as the main interface to register a new user.

3. Create processUser.jsp page to process the record.

4. Create login.jsp page to login to the system.

5. Create doLogin.jsp to validate username and password. If validation is successful, redirect the page to main.jsp page that displays the username, firstname and lastname.

6. If validation is unsuccessful, redirect the page to login.jsp with message 'Invalid username or password..!'

**Coding:**

**Ex.insertUser.htm**

```
<!DOCTYPE html>

<!--
Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt
to change this license
Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Html.html to
edit this template
-->

<html>
<head>
<title>Register New User</title>
</head>
<body>
<h2>Register New User</h2>
<form action="ex.processUser.jsp" method="post">
<label for="username">Username:</label>
<input type="text" id="username" name="username" maxlength="15"
required><br><br>
<label for="password">Password:</label>
<input type="password" id="password" name="password" maxlength="10"
required><br><br>
<label for="firstname">First Name:</label>
<input type="text" id="firstname" name="firstname" maxlength="50"
required><br><br>
<label for="lastname">Last Name:</label>
<input type="text" id="lastname" name="lastname" maxlength="50"
```

```
required><br><br>
<input type="submit" value="Register">
</form>
</body>
</html>
```

### **ex.processUser.jsp**

```
<%--
    Document   : ex.processUser
    Created on : 15 Jun 2024, 15:53:02
    Author      : Atirah s67397
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ page import="java.sql.*" %>
<%
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    String firstname = request.getParameter("firstname");
    String lastname = request.getParameter("lastname");

    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csm3023", "root",
"admin");

        String sql = "INSERT INTO userprofile2 (username, password, firstname,
lastname) VALUES (?, ?, ?, ?)";
        PreparedStatement stmt = conn.prepareStatement(sql);
```

```
stmt.setString(1, username);
stmt.setString(2, password);
stmt.setString(3, firstname);
stmt.setString(4, lastname);

int rowsInserted = stmt.executeUpdate();
if (rowsInserted > 0) {
    out.println("A new user was inserted successfully!");
}

stmt.close();
conn.close();
} catch (Exception e) {
    e.printStackTrace();
}
%>
```

### Ex.login.jsp

```
<%--
    Document : ex.login
    Created on : 15 Jun 2024, 15:53:34
    Author : User
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <title>Login</title>
</head>
<body>
    <h2>Login</h2>
    <form action="ex.doLogin.jsp" method="post">
```

```

        <label for="username">Username:</label>
        <input type="text" id="username" name="username" maxlength="15"
required><br><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"
maxlength="10" required><br><br>
        <input type="submit" value="Login">
    </form>
    <%
        if (request.getAttribute("errorMessage") != null) {
            out.println("<p style='color:red'>" +
request.getAttribute("errorMessage") + "</p>");
        }
    %>
</body>
</html>

```

### Ex.dologin.jsp

```

<%--
    Document : ex.doLogin
    Created on : 15 Jun 2024, 15:54:47
    Author : Atirah s67397
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ page import="java.sql.*" %>
<%
    String username = request.getParameter("username");
    String password = request.getParameter("password");

    boolean isValidUser = false;
    String firstname = "";

```

```

String lastname = "";

try {
    Class.forName("com.mysql.jdbc.Driver");
    Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csm3023", "root",
"admin");

    String sql = "SELECT firstname, lastname FROM userprofile2 WHERE
username = ? AND password = ?";
    PreparedStatement stmt = conn.prepareStatement(sql);
    stmt.setString(1, username);
    stmt.setString(2, password);

    ResultSet rs = stmt.executeQuery();
    if (rs.next()) {
        isValidUser = true;
        firstname = rs.getString("firstname");
        lastname = rs.getString("lastname");
    }

    rs.close();
    stmt.close();
    conn.close();
} catch (Exception e) {
    e.printStackTrace();
}

if (isValidUser) {
    request.setAttribute("username", username);
    request.setAttribute("firstname", firstname);
    request.setAttribute("lastname", lastname);
    RequestDispatcher dispatcher =

```

```

request.getRequestDispatcher("ex.main.jsp");
    dispatcher.forward(request, response);
} else {
    request.setAttribute("errorMessage", "Invalid username or password..!");
    RequestDispatcher dispatcher =
request.getRequestDispatcher("ex.login.jsp");
    dispatcher.forward(request, response);
}
%>

```

### Ex.main.jsp

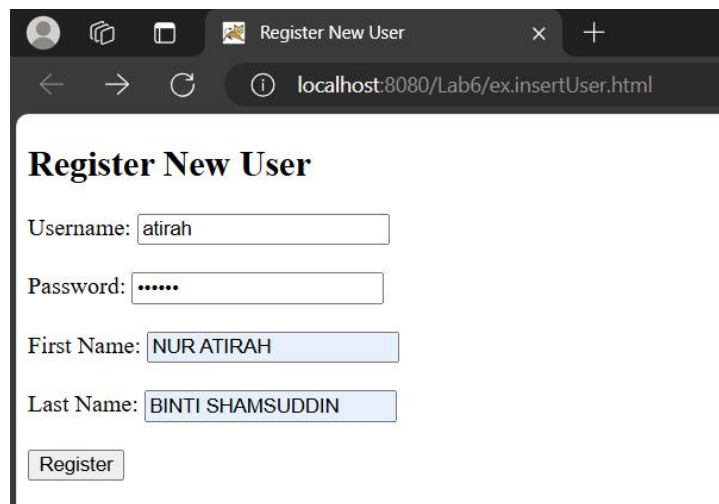
```

<%--
    Document : ex.main
    Created on : 15 Jun 2024, 15:55:22
    Author : Atirah s67397
--%>
<%@ page contentType="text/html; charset=UTF-8" %>
<%
    String username = (String) request.getAttribute("username");
    String firstname = (String) request.getAttribute("firstname");
    String lastname = (String) request.getAttribute("lastname");
%>
<!DOCTYPE html>
<html>
<head>
    <title>Main Page</title>
</head>
<body>
    <h2>Welcome, <%= firstname %> <%= lastname %>!</h2>
    <p>Your username: <%= username %></p>
</body>
</html>

```



Output:



Register New User

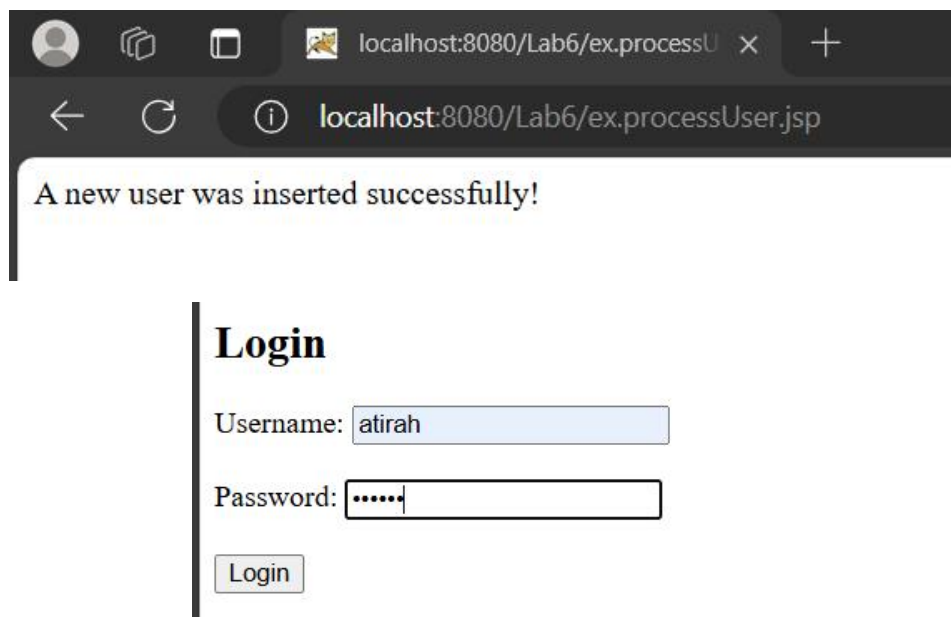
Username:

Password:

First Name:

Last Name:

VALID



localhost:8080/Lab6/ex.insertUser.html

localhost:8080/Lab6/ex.processUser.jsp

A new user was inserted successfully!

Login

Username:

Password:

**Welcome, NUR ATIRAH BINTI SHAMSUDDIN!**

Your username: atirah

INVALID

## Login

Username:

Password:

Invalid username or password..!