

## CASESTUDY-5

### **Aim:**

Develop a Spring Web MVC application where Student can enter his/her details like UserName, Roll Number, Branch Name, College Name in an User Interface, after clicking on submit button it has to store data into the Database and display all the details below to the submit button in a table.

### **Program:**

#### **Index.jsp**

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Welcome</title>
```

```
    <style>
```

```
        body {
```

```
            font-family: Arial, sans-serif;
```

```
            background-color: #f0f8ff;
```

```
            text-align: center;
```

```
            margin: 0;
```

```
            padding: 0;
```

```
        }
```

```
        h1 {
```

```
            color: #333;
```

```
        }
```

```
        a {
```

```
            text-decoration: none;
```

```
            color: #ffffff;
```

```
            background-color: #007bff;
```

```
            padding: 10px 20px;
```

```
            border-radius: 5px;
```

```

    }
    a:hover {
        background-color: #0056b3;
    }
</style>
</head>
<body>
    <h1>22251A0574</h1>
    <h1>Spring MVC Application for Student Records</h1>
    <p><a href="student.htm">Enter your Details</a></p>
</body>
</html>

```

### **student.jsp**

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
    <title>Student Details</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f9f9f9;
            margin: 20px;
        }
        h1, h2 {
            color: #444;
            text-align: center;
        }
    
```

```
form {  
    margin: 0 auto;  
    width: 300px;  
    padding: 20px;  
    background-color: #ffffff;  
    border: 1px solid #ccc;  
    border-radius: 8px;  
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);  
}  
  
input[type="text"], input[type="submit"] {  
    width: 100%;  
    padding: 10px;  
    margin: 10px 0;  
    border: 1px solid #ccc;  
    border-radius: 5px;  
}  
  
input[type="submit"] {  
    background-color: #007bff;  
    color: #ffffff;  
    cursor: pointer;  
}  
  
input[type="submit"]:hover {  
    background-color: #0056b3;  
}  
  
table {  
    margin: 20px auto;  
    border-collapse: collapse;  
    width: 80%;  
}
```

```

table, th, td {
    border: 1px solid #ddd;
}
th, td {
    padding: 10px;
    text-align: left;
}
th {
    background-color: #007bff;
    color: white;
}
tr:nth-child(even) {
    background-color: #f2f2f2;
}
</style>
</head>
<body>
<h1>Student Management</h1>
<form method="post">
    <label for="rollNumber">Roll Number:</label>
    <input type="text" id="rollNumber" name="rollNumber" required/><br>
    <label for="userName">User Name:</label>
    <input type="text" id="userName" name="userName" required/><br>
    <label for="branchName">Branch Name:</label>
    <input type="text" id="branchName" name="branchName" required/><br>
    <label for="collegeName">College Name:</label>
    <input type="text" id="collegeName" name="collegeName" required/><br>
    <input type="submit" name="submit" value="Submit and Save" />
    <br>${msg}

```

</form>

<h2>All Students</h2>

<table>

<thead>

<tr>

<th>Roll Number</th>

<th>User Name</th>

<th>Branch Name</th>

<th>College Name</th>

</tr>

</thead>

<tbody>

<c:forEach var="student" items="\${students}">

<tr>

<td>\${student.rollNumber}</td>

<td>\${student.userName}</td>

<td>\${student.branchName}</td>

<td>\${student.collegeName}</td>

</tr>

</c:forEach>

</tbody>

</table>

</body>

</html>

### **dispatcher-servlet.xml**

<?xml version='1.0' encoding='UTF-8' ?>

<!-- was: <?xml version="1.0" encoding="UTF-8"?> -->

<beans xmlns="http://www.springframework.org/schema/beans"

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:p="http://www.springframework.org/schema/p"
xmlns:aop="http://www.springframework.org/schema/aop"
xmlns:tx="http://www.springframework.org/schema/tx"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-4.0.xsd
http://www.springframework.org/schema/aop
http://www.springframework.org/schema/aop/spring-aop-4.0.xsd
http://www.springframework.org/schema/tx
http://www.springframework.org/schema/tx/spring-tx-4.0.xsd">

```

```

<bean
class="org.springframework.web.servlet.mvc.support.ControllerClassNameHandlerMapping"
/>

```

```

<bean class="controller.StudentController"/>

```

```

<!--

```

Most controllers will use the ControllerClassNameHandlerMapping above, but for the index controller we are using ParameterizableViewController, so we must define an explicit mapping for it.

```

-->

```

```

<bean id="urlMapping"
class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">
    <property name="mappings">
        <props>
            <prop key="index.htm">indexController</prop>
        </props>
    </property>
</bean>

```

```

<bean id="viewResolver"
    class="org.springframework.web.servlet.view.InternalResourceViewResolver"
    p:prefix="/WEB-INF/jsp/"

```

```

        p:suffix=".jsp" />
<!--
    The index controller.
-->
<bean name="indexController"
        class="org.springframework.web.servlet.mvc.ParameterizableViewController"
        p:viewName="index" />
</beans>

```

### **StudentController.java**

```
package controller;
```

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import model.Student;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.AbstractController;

```

```
public class StudentController extends AbstractController {
```

```
    @Override
```

```
    protected ModelAndView handleRequestInternal(HttpServletRequest request,
        HttpServletResponse response) throws Exception {
```

```
        ModelAndView mv = new ModelAndView("student");
```

```

// Check if the form is submitted
if (request.getParameter("submit") != null) {

    String rollNumber = request.getParameter("rollNumber");

    String userName = request.getParameter("userName");

    String branchName = request.getParameter("branchName");

    String collegeName = request.getParameter("collegeName");


    // Insert student into the database

    try {

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

        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/db574", "root",
"Harshitha@1056");

        Statement statement = con.createStatement();

        String sql = "INSERT INTO students574 (roll_number, user_name, branch_name,
college_name) " +
            "VALUES ('" + rollNumber + "', '" + userName + "', '" + branchName + "', '" +
collegeName + "')";

        int result = statement.executeUpdate(sql);

        if (result > 0) {

            mv.addObject("msg", "Saved Successfully");

        }

    } catch (Exception e) {

        mv.addObject("msg", "Error: " + e.getMessage());

    }

}


// Fetch all students

List<Student> students = new ArrayList<>();

try {

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

```



```

        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/db574", "root",
"Harshitha@1056");

        Statement statement = con.createStatement();

        String sql = "SELECT * FROM students574";

        ResultSet rs = statement.executeQuery(sql);

        while (rs.next()) {

            Student student = new Student();

            student.setRollNumber(rs.getString("roll_number"));

            student.setUserName(rs.getString("user_name"));

            student.setBranchName(rs.getString("branch_name"));

            student.setCollegeName(rs.getString("college_name"));

            students.add(student);

        }

    } catch (Exception e) {

        mv.addObject("msg", "Error fetching students: " + e.getMessage());

    }

    mv.addObject("students", students);

    return mv;

}

}

```

### **Student.java**

```

package model;

public class Student {

    private String rollNumber;

    private String userName;

    private String branchName;

    private String collegeName;

```

// Getters and Setters

```
public String getRollNumber() {  
    return rollNumber;  
}
```

```
public void setRollNumber(String rollNumber) {  
    this.rollNumber = rollNumber;  
}
```

```
public String getUsername() {  
    return userName;  
}
```

```
public void setUsername(String userName) {  
    this.userName = userName;  
}
```

```
public String getBranchName() {  
    return branchName;  
}
```

```
public void setBranchName(String branchName) {  
    this.branchName = branchName;  
}
```

```
public String getCollegeName() {  
    return collegeName;  
}
```

```

public void setCollegeName(String collegeName) {

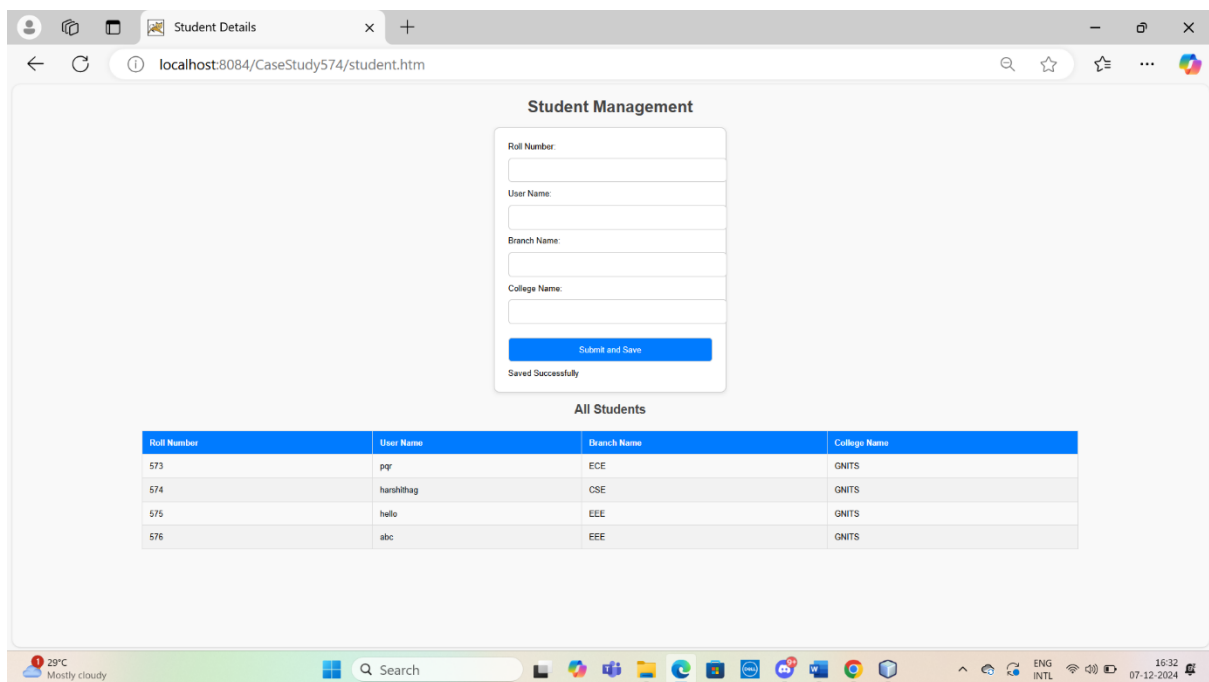
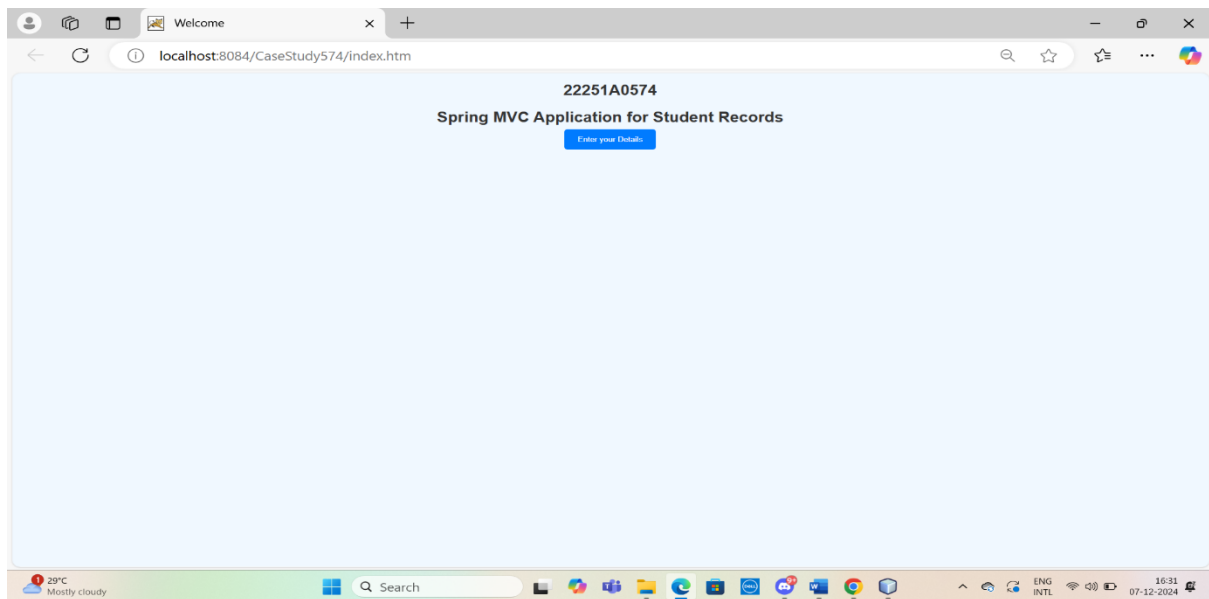
    this.collegeName = collegeName;

}

}

```

## Output:



```
mysql> use db574;
```

```
Database changed
```

```
mysql> select * from students574;
```

roll_number	user_name	branch_name	college_name
573	pqr	ECE	GNITS
574	harshithag	CSE	GNITS
575	hello	EEE	GNITS
576	abc	EEE	GNITS

```
4 rows in set (0.00 sec)
```