

PRACTICAL 6(C) – JSP STUDENTMASTER CRUD OPERATIONS

AIM

To design and develop a JSP-based web application to perform CRUD (Create, Read, Delete) operations on StudentMaster table containing RollNo, Name, Semester, and Course.

SOFTWARE REQUIREMENTS

- Operating System: Windows / Linux
- JDK: Version 7 or 8
- NetBeans IDE 7.x
- Apache Tomcat Server 7.x
- MySQL Database

THEORY

CRUD operations refer to Create, Read, Update, and Delete functionalities used in database-driven applications. JSP works with JDBC to connect to relational databases and perform these operations. In this practical, JSP pages are used to insert, delete, and display records from the StudentMaster table.

DATABASE DESIGN

Database Name: studentdb

Table Name: studentmaster

Columns:

- rollno (INT, Primary Key)
- name (VARCHAR(50))
- semester (INT)
- course (VARCHAR(50))

SOURCE CODE – index.jsp (Student Form)

```
<html>
<body>
<h2>Student Registration</h2>
<form action="insert.jsp" method="post">
Roll No: <input type="text" name="rollno"><br><br>
Name: <input type="text" name="name"><br><br>
Semester: <input type="text" name="semester"><br><br>
Course: <input type="text" name="course"><br><br>
<input type="submit" value="Add Student">
</form>
<br>
<a href="view.jsp">View Students</a>
</body>
</html>
```

SOURCE CODE – insert.jsp (Insert Record)

```
<%@ page import="java.sql.*" %>
<%
int rollno = Integer.parseInt(request.getParameter("rollno"));
String name = request.getParameter("name");
int semester = Integer.parseInt(request.getParameter("semester"));
String course = request.getParameter("course");

Class.forName("com.mysql.jdbc.Driver");
Connection con = DriverManager.getConnection(
"jdbc:mysql://localhost:3306/studentdb","root","");
PreparedStatement ps = con.prepareStatement(
"insert into studentmaster values(?, ?, ?, ?)");

ps.setInt(1, rollno);
ps.setString(2, name);
ps.setInt(3, semester);
ps.setString(4, course);

ps.executeUpdate();
out.println("Student Record Inserted Successfully");
%>
<a href="index.jsp">Back</a>
```

SOURCE CODE – view.jsp (Display Records)

```
<%@ page import="java.sql.*" %>
<html>
<body>
<h2>Student Records</h2>
<table border="1">
<tr>
<th>Roll No</th>
<th>Name</th>
<th>Semester</th>
<th>Course</th>
<th>Action</th>
</tr>
<%
Class.forName("com.mysql.jdbc.Driver");
Connection con = DriverManager.getConnection(
"jdbc:mysql://localhost:3306/studentdb","root","");
Statement st = con.createStatement();
ResultSet rs = st.executeQuery("select * from studentmaster");

while(rs.next()) {
%>
<tr>
<td><%= rs.getInt(1) %></td>
<td><%= rs.getString(2) %></td>
<td><%= rs.getInt(3) %></td>
<td><%= rs.getString(4) %></td>
<td>
<a href="delete.jsp?rollno=<%= rs.getInt(1) %>">Delete</a>
</td>
</tr>
<%
}
%>
</table>
<br>
<a href="index.jsp">Add New Student</a>
</body>
</html>
```

SOURCE CODE – delete.jsp (Delete Record)

```

<%@ page import="java.sql.*" %>
<%
int rollno = Integer.parseInt(request.getParameter("rollno"));

Class.forName("com.mysql.jdbc.Driver");
Connection con = DriverManager.getConnection(
"jdbc:mysql://localhost:3306/studentdb", "root", "");

PreparedStatement ps = con.prepareStatement(
"delete from studentmaster where rollno=?");

ps.setInt(1, rollno);
ps.executeUpdate();

out.println("Student Record Deleted Successfully");
%>
<a href="view.jsp">Back to List</a>

```

NETBEANS 7 – DETAILED EXECUTION STEPS

1. Open NetBeans IDE 7
2. File → New Project → Java Web → Web Application
3. Project Name: StudentMasterApp
4. Select Apache Tomcat Server
5. Click Finish
6. Create database studentdb and table studentmaster in MySQL
7. Add MySQL Connector JAR to Libraries
8. Create JSP files: index.jsp, insert.jsp, view.jsp, delete.jsp
9. Paste respective code in each JSP file
10. Right-click project → Run

OUTPUT

- Student records are added successfully
- Records are displayed in tabular format
- Records can be deleted using Delete link

RESULT

Thus, CRUD operations on StudentMaster table were successfully implemented using JSP and JDBC.