Practical 11:

Develop a shopping cart using servlet HttpSession object which provides the facility to select items from various categories (like mobile, laptop, etc.), add them to cart, view the cart and edit the items added in the shopping cart.

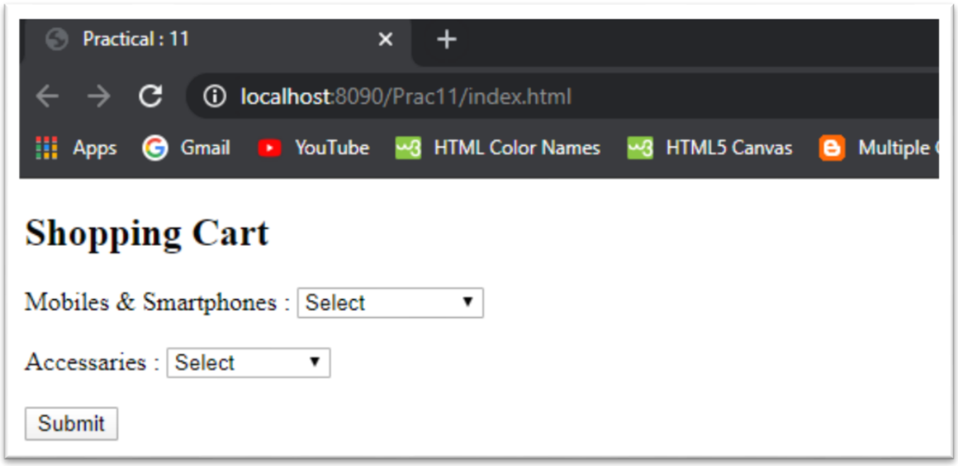
Source Code (Shopping Cart.java) :

```
import java.io.*;
import javax.servlet.annotation.WebServlet;
import javax.servlet.*;
@WebServlet("/ShoppingCart")
class ShoppingCart extends HttpServlet {
    protected void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException,
IOException {
                PrintWriter out=res.getWriter();
                HttpSession session=req.getSession(false);
                out.println("<!DOCTYPE html>\r\n" +
                                        "<html>\r\n" +
                                        "<head>\r\n" +
                                        "<title>Practical : 11</title>\r\n" +
                                        "</head>\r\n" +
                                        "<body>\r\n" +
"<h1>Update cart</h1> "
                                        +"<form action='submitted_form' method='Post' target='_blank'>\r\n" +
                                        "           <fieldset>\r\n" +
                                        "                                       <label>Electronics</label>\r\n" +
                                        "                                       <select name='Electronics id='Electronics'>\r\n" +
                                        "                                         <option>Please select</option>\r\n" +
                                        "                                                  <option value='Mobile'>Mobile</option>\r\n"
+
                                        "                                                  <option
value='TelePhone'>TelePhone</option>\r\n" +
                                        "                                                  <option
value='Headphone'>Headphone</option>\r\n" +
                                        "               <option value='Earphone'>Earphone</option>\r\n" +
                                        "               <option value='Ipad'>Ipad</option>\r\n" +
                                        "                                       </select>\r\n" +
                                        "            <filedset>\r\n" +
                                        "     <input type='submit' value='Add to cart'>\r\n" +
                                        "     </form>\r\n" +
                                        " <script type="text/javascript">\r\n" +
                                        "  var x=document.getElementById('Electronics').options;\r\n" +
                                        " (for let i=0;i<x.length;i++)\r\n"  +
                                        " {\r\n" +
                                         "  if(x[i].value==\""+session.getAttribute("Electronics")+"\")\r\n" +
                                        " {\r\n"+
                                        "                   document.getElementById('Electronics').selectedIndex=i;\r\n"
+
                                        " }\r\n" +
                                        " }\r\n" +
                                        " </script>\r\n"  +
                                        "  </body>\r\n"  +
                                        "</html>");
    }
     protected void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException,
IOException {
                res.setContentType("text/html");
                PrintWriter out=res.getWriter();
                HttpSession session=req.getSession();
                session.setAttribute("Electronics", req.getParameter("Electronics"));
                out.println("Electronics:"+session.getAttribute("Electronics")+"</br>");
                out.println("<a href='submitted_form'>Edit cart:</a>");
    }
}
```

Source Code (Index File) :

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Practical : 11</title>
</head>
<body>
        <form action="buy">
                <h2>Shopping Cart</h2>
                Mobiles & Smartphones :
                <select name="mobiles">
                        <option value="none">Select</option>
                        <option value="IPhone 11">IPhone 11</option>
                        <option value="OnePlus 3T">OnePlus 3T</option>
                        <option value="Galaxy J 7 Max">Galaxy J 7 Max</option>
                </select><br /><br />
                Accessaries :
                <select name="Accessaries">
                <option value="none">Select</option>
                <option value="Fast Charger">Fast Charger</option>
                <option value="Air Buds">Air Buds</option>
                <option value="Case Cover">Case Cover</option>
            </select><br /><br />
            <input type="submit">
        </form>
</body>
</html>
```

Output :

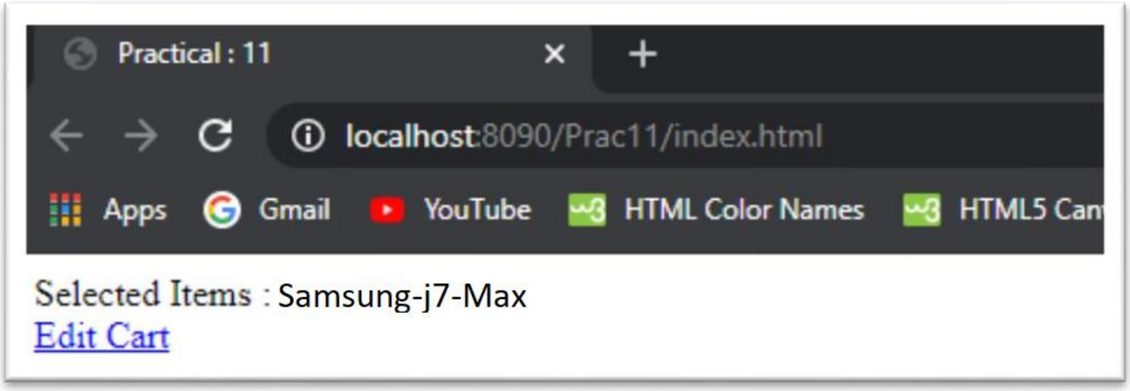


Shopping Cart

Mobiles & Smartphones : Select ▾

Accessaries : Select ▾

Submit



Selected Items : Samsung-j7-Max
Edit Cart

Practical 12:

Implement a servlet logging in a user. Use filter that authenticates the login info of the user. If the credentials are correct, it takes the user to welcome page and creates cookies of the user else asks to re-enter the details.

Source Code (Login.java) :

```
import java.io.*;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
@WebServlet("/Login")
class Login extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                PrintWriter out= response.getWriter();
                RequestDispatcher rd;
                String Login=request.getParameter("Login");
                String pwd=request.getParameter("pwd");
                Cookie cookie = new Cookie("Login",Login);
                Cookie cookie1 = new Cookie("pwd",pwd);
                response.addCookie(cookie);
                response.addCookie(cookie1);
                if(Login.equals("UserName") && pwd.equals("Password"))
                {
                        rd=request.getRequestDispatcher("Welcome");
                        rd.forward(request, response);
                }
                else
                {
                        out.println("<p><h1>Incorrect Login ID/Password</h1></p>");
                        out.println("<p><h1>Re-Enter Details</h1></p>");
                        rd=request.getRequestDispatcher("/1.html");
                        rd.include(request, response);
                }
    }
}
```

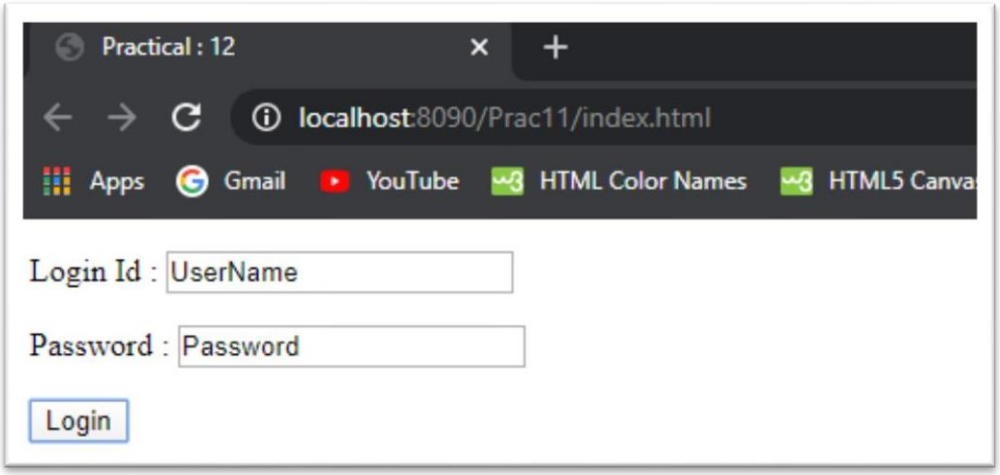
Source Code (Welcome.java) :

```
import java.io.*;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
@WebServlet("/Welcome")
class Welcome extends HttpServlet {
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
         PrintWriter out=response.getWriter();
         String username=request.getParameter("Login");
         out.println("<h1>"+"Welcome "+username+"</h1>");
  }
}
```

Source Code (index.html) :

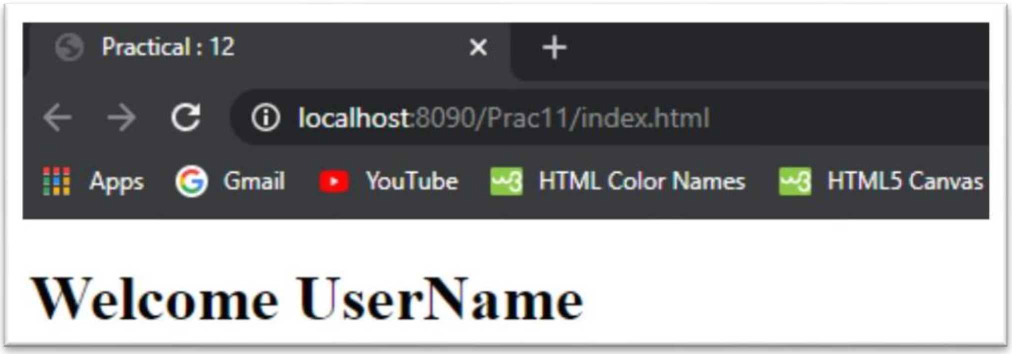
```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Practical : 12</title>
</head>
<body>
<form action="welcome" method="post">
 <p>Login Id : <input type="text" name="login"></p>
 <p>Password : <input type="text" name="pwd"></p>
 <p><input type="submit" value="Login"></p>
 </form>
</body>
</html>
```

Output :



Login Id : UserName

Password : Password



Welcome UserName

Practical 13:

Implement a servlet that adds two numbers. Use two filters in a chain mapped to a servlet such that the first filter checks the number for a valid format, i.e. numbers only contain digits and the second filter checks them for a range (1 to 100). If both conditions are fulfilled then only the servlet performs the operation and displays the result; otherwise, the execution ends with an appropriate message from the respective filter.

Source Code (Add.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<title>Practical : 13</title>
</head>
<body>
<form action="<%=request.getContextPath() %>/AddNumbers" method="post">

<%String err=(String)session.getAttribute("Error");
     if(err!=null){%>

<span style="color:red"><%=err%></span>
<%session.removeAttribute("Error");
}%>  <br/>

 <input type="text" name="first"> + <input type="text" name="second"> = ${sum}
<%session.removeAttribute("sum");%> <input type="submit">

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

Source Code (Validator.java) :

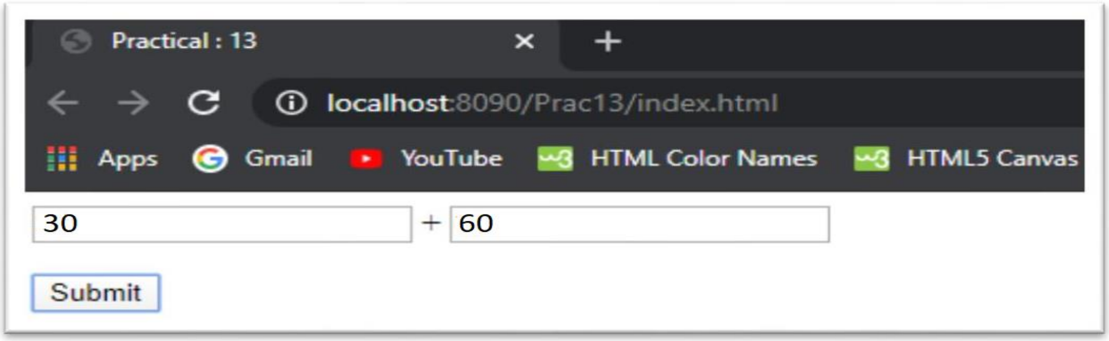
```
import java.io.IOException;
import javax.servlet.*;
@WebFilter("/AddNumbers")
```

```
class Validator implements Filter {
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
throws IOException, ServletException {
        HttpServletRequest req=(HttpServletRequest)request;
        HttpServletResponse res=(HttpServletResponse)response;
        try{
                int  a=Integer.parseInt(request.getParameter("first"));
                int b=Integer.parseInt(request.getParameter("second"));
                chain.doFilter(request, response);
        }catch (NumberFormatException e) {
                HttpSession session=req.getSession();
                session.setAttribute("Error","Please enter number in valid format");
                res.sendRedirect("addNumber.jsp");  }
        }
        @Override
        public void destroy(){}
        @Override
        public void init(FilterConfig arg0) throws ServletException {}
}
```

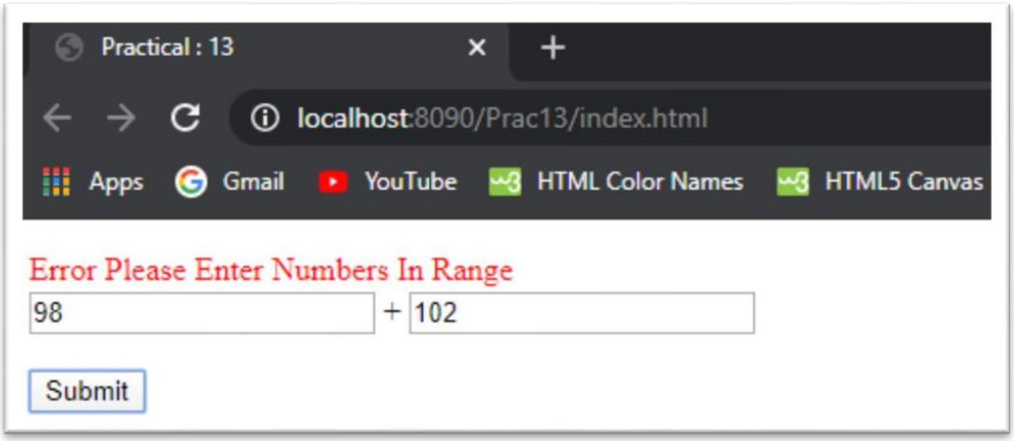
Source Code (CheckRange.java) :

```
import java.io.IOException;
import javax.servlet.*;
@WebFilter("/AddNumbers")
class CheckRange implements Filter {
public void destroy(){}
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
throws IOException, ServletException {
        HttpServletRequest req=(HttpServletRequest)request;
        HttpServletResponse res=(HttpServletResponse)response;
        String first=request.getParameter("first");
        String second=request.getParameter("second");
        try{
                int  a=Integer.parseInt(first);
                int b=Integer.parseInt(second);
                if(!(a>=1&&a<=100&&b>=1&&b<=100)){
                        HttpSession session=req.getSession();
                        session.setAttribute("Error","Please enter number in range 1 to 100");
                        res.sendRedirect("addNumber.jsp");
                }else   chain.doFilter(request, response);
        }catch(Exception e){chain.doFilter(request, response);}
        }
        @Override
        public void init(FilterConfig arg0) throws ServletException{}
}
```

Output :



30 + 60



Error Please Enter Numbers In Range

98 + 102

Practical 14:

Develop a Web application that prepares student mark sheet using marks obtained for each subject. Develop the application using JSP, JDBC and JavaBeans such that the marks stored in the database are accessed using JDBC by the logic implemented in JavaBeans and the presentation is prepared using JSP which accesses the JavaBeans.

Source Code (Menu.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<title>Practical : 14</title>
</head>
<body>
<a href="takeDetail.jsp">Insert Student Detail</a><br><br>
<a href="enterEnrollment.jsp">Search Student</a>
</body>
</html>
```

Source Code (TakeDetails.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<title>Practical : 14</title>
</head>
<body>
<form action="<%=request.getContextPath()%>/MarksheetServlet" method="post">
        Enter details:<br><br>
        Enrollment : <input type="text" name="en" ><br><br>
        Name : <input type="text" name="name"><br><br>
        IP Marks : <input type="text" name="ip"><br><br>
        AJ Marks : <input type="text"  name="aj"><br><br>
        WT Marks : <input type="text" name="wt"><br><br>
        DCDR Marks : <input type="text" name="dcdr"><br><br>
        <input type="hidden" name="key" value="insert">
        <input type="submit" >
</form>
</body>
</html>
```

Source Code (AddEnrollment.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<html>
<head>
<title> Practical : 14</title>
</head>
<body>
<form action="<%=request.getContextPath()%>/MarksheetServlet" method="post">
Enrollment : <input type="text" name="en">
<input type="hidden" name="key" value="searchMarksheet">
<input type="submit" value="search">
</form>
</body>
</html>
```

Source Code (ViewMarks.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<%@taglib prefix="c" uri="http://java.sun.com/jstl/core_rt" %>
<!DOCTYPE html>
<html>
<head>
<title>Insert title here</title>
</head>
<body>
<c:set var="vo" value="${sessionScope.vo}"></c:set>
Name : ${vo.name}&nbsp&nbsp&nbsp&nbsp&nbspEnrollment : ${vo.enrollment}<br><br>
<table border="1" style="width: 300px ; border-collapse: coolapse" >
 <tr>
   <td>SUBJECT</td>
   <td>MARK</td>
 </tr>
 <tr>
   <td>IP</td>
```

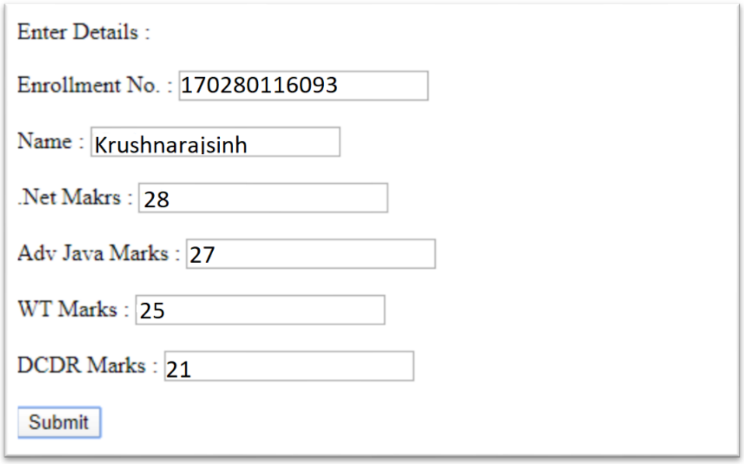
```
  <td>${vo.ipMarks}</td>
 </tr>
 <tr>
  <td>DCDR</td>
  <td>${vo.dcdrMarks}</td>
 </tr>
 <tr>
  <td>WT</td>
  <td>${vo.wtMarks}</td>
 </tr>
 <tr>
  <td>AJ</td>
  <td>${vo.ajMarks}</td>
 </tr>
</table>
</body>
</html>
```

Source Code (MarksView.java) : JavaBean

```
package javaBeans;
public class MarksheetVO {
	private String enrollment;
	private String name;
	private int ipMarks, ajMarks, wtMarks, dcdrMarks;
	public String getEnrollment() {
		return enrollment;
	}
	public void setEnrollment(String enrollment) {
		this.enrollment = enrollment;
	}
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public int getIpMarks() {
		return ipMarks;
	}
	public void setIpMarks(int ipMarks) {
		this.ipMarks = ipMarks;
	}
	public int getAjMarks() {
		return ajMarks;
	}
	public void setAjMarks(int ajMarks) {
		this.ajMarks = ajMarks;
	}
```

```
    public int getWtMarks() {
            return wtMarks;
    }
    public void setWtMarks(int wtMarks) {
            this.wtMarks = wtMarks;
    }
    public int getDcdrMarks() {
            return dcdrMarks;
    }
    public void setDcdrMarks(int dcdrMarks) {
            this.dcdrMarks = dcdrMarks;
    }  }
```

Output :



Enter Details :

Enrollment No. : 170280116093

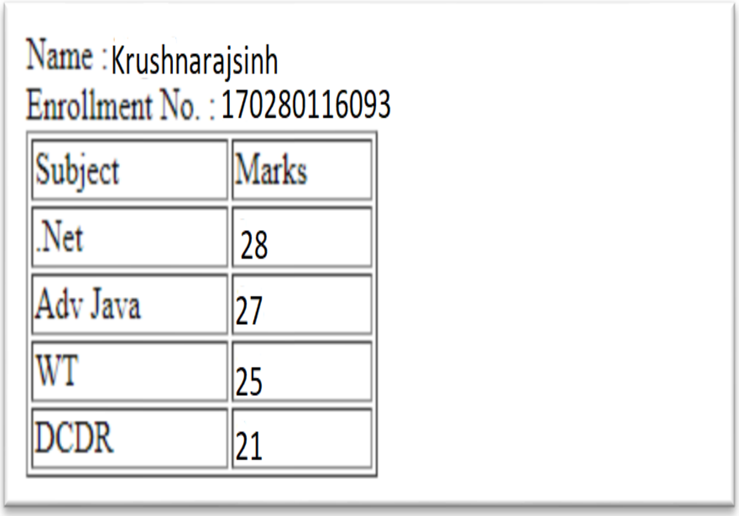
Name : Krushnarajsinh

.Net Makrs : 28

Adv Java Marks : 27

WT Marks : 25

DCDR Marks : 21



Name :Krushnarajsinh
Enrollment No. : 170280116093

Subject	Marks
.Net	28
Adv Java	27
WT	25
DCDR	21

Practical 15:

Apply MVC approach using JSP to develop an application that displays bank account balances. The controller servlet reads a customer ID and passes that to some data-access code that returns a BankCustomer value bean. The servlet then stores the bean in the HttpServletRequest object where it will be accessible from destination JSP pages but nowhere else. If the account balance of the resulting customer is negative, the servlet forwards to a page designed for delinquent customers. If the customer has a positive balance of less than $10,000, the servlet transfers to the standard balance display page. Next, if the customer has a balance of $10,000 or more, the servlet forwards the request to a page reserved for elite customers. Finally, if the customer ID is unrecognized, an error page is displayed.

Source Code (index.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<title>Student Marks Entry</title>
</head>
<body>
        <h1>Enter credentials</h1>
        <form action='submitted_form' method='POST' target='_blank'>
                <input type="text" name="uname" placeholder="User name" /> <br
                        /> <input type="password" name="passwd"
                        placeholder="Password" /><br /> <input type='submit'
                        value='Submit'></form>
</body>
</html>
```

Source Code (DBAccBean.java) :

```
import java.sql.*;
        public class DBAccBean {
          private int aid;
          private String uname;
          private String upass;
          private double ubal;
          public Connection conn = null;
          private String url = "";
          public ResultSet rs = null;
           public DbAccBean() throws ClassNotFoundException {
                   try {
                             Class.forName("com.mysql.jdbc.Driver");
                             url="mysql//jdbc//localhost:1234/DB_01";
                             conn = DriverManager.getConnection(url);
                   }      catch      (SQLException      e)      {
                            System.out.println(e.getMessage());
                   }
          }
         public void setaid(int aid) {
                   this.aid = aid;
                   }
          public int getaid() {
                   return aid;
           }
          public void setuname(String uname) {
                   this.uname = uname;
          }
          public String getuname() {
                   return uname;
          }
          public void setupass(String upass) {
                   this.upass = upass;
          }
          public String getupass() {
                          return upass;
          }
          public void setubal(double ubal) {
                                   this.ubal = ubal;
          }
          public double getubal() {
                   return ubal;
          }
        public void insert() {
              try {
```

```
            String sql = "INSERT INTO account values('" + aid + "','" + uname + "','"
            + upass + "','"
+ ubal + "');";
            Statement stmt = conn.createStatement();
            stmt.executeUpdate(sql);
        } catch (SQLException e) {
            System.out.println(e.get
            Message());
        } catch (Exception e) {
            System.out.println(e.get
            Message());
        }
    }
    public ResultSet show_all() {
        try {
            String sql = "SELECT * FROM account";
            Statement stmt = conn.createStatement();
            rs = stmt.executeQuery(sql);
        } catch (SQLException e) {
            System.out.println(e.get
            Message());

        } catch (Exception e) {
            System.out.println(e.get
            Message());
        }
        return rs;
    }
    public int[] Login_verify(String uname, String upass) {
        int flag = 0;
        int bal;
        int[] arr = new int[2];
        try {
            String sql = "SELECT * FROM account";
            Statement stmt = conn.createStatement();
            rs = stmt.executeQuery(sql);
            while (rs.next()) {
                if (rs.getString("uname").equals(uname)) {
                    if (rs.getString("upass").equals(upass)) {
                        flag = Integer.parseInt(rs.getString("aid"));
                        bal = Integer.parseInt(rs.getString("ubal"));
                        arr[0] = flag;
                        arr[1] = bal;
                        return arr;
                    } else {
                        arr[0] = -2;
                        arr[1] = -1;
                         return arr;
                    }
                }
            }
        } catch (SQLException e) {
            System.out.println(e.get
            Message());

        } catch (Exception e) {
            System.out.println(e.get
            Message());
        }
        arr[0] = -1;
        arr[1] = -1;
        return arr;
}}
```

Source Code (Login.java) :

```
import java.io.*;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
@WebServlet("/Login")
class Login extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
response.getWriter().append("Served at: ").append(request.getContextPath());
}
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        int[] arr=new int[2];
        try {
            arr=new
DbAccBean().Login_verify(request.getParameter("uname"),request.getParameter("passwd"));
            if(arr[0]==-2)
            {
                request.setAttribute("mess","Correct id but invalid
                Password<br/>"); request.setAttribute("ubal","");
```

```
                        RequestDispatcher
                        rd=request.getRequestDispatcher("error.jsp");
                        rd.forward(request, response);
                            }
                            else if(arr[1]==-1)
                            {
                             request.setAttribute("mess","Invalid Credentials<br/>");
                                     request.setAttribute("ubal","");
   RequestDispatcher rd=request.getRequestDispatcher("error.jsp");
rd.forward(request, response);
                            }
                            else
                            {
                                      request.setAttribute("mess","Logged In<br/>");
                                      request.setAttribute("ubal",arr[1]); if(arr[1]<0)
                                      {
                            RequestDispatcher rd=request.getRequestDispatcher("delinquent.jsp");
                            rd.forward(request, response);
                                      }
                                      else if(arr[1]<10000)
                                      {
                                                RequestDispatcher
                                                rd=request.getRequestDispatcher("regular.jsp");
                                                rd.forward(request, response);
                                      }
                                      else
                                      {
                                      RequestDispatcher
                            rd=request.getRequestDispatcher("elite.jsp"); rd.forward(request,
                            response);
                                      }
                            }
                  }
          catch(ClassNotFoundException e)
          {
                    e.printStackTrace();
          out.println(e.getMessage());
          }
          }
}
```

Source Code (Error.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
</head>
<body>

        <h1 style="color:red;"> Error Has Occurred!</h1>
        <h3><%= request.getAttribute("mess") %></h3>
</body>
</html>
```

Source Code (Delinquent.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
</head>
<body>
        <h1><%= request.getAttribute("mess") %></h1>
        <h3 style="color:red;">Balance:<%=
        request.getAttribute("ubal") %></h3> <h3>You are a
        delinquent customer</h3>
</body>
</html>
```

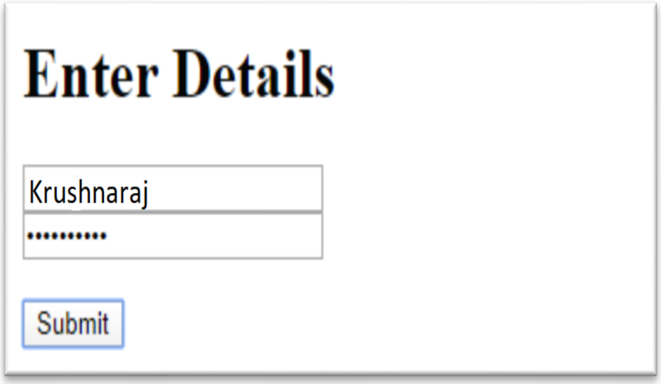
Source Code (Ellite.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
</head>
<body>
        <h1><%= request.getAttribute("mess") %></h1>
        <h3 style="color:purple;">Balance:<%=
        request.getAttribute("ubal") %></h3> <h3>You are a Elite
        customer</h3>
</body>
</html>
```

Source Code (Regular.jsp) :

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
</head>
<body>
        <h1><%= request.getAttribute("mess") %></h1>
        <h3 style="color:green;">Balance:<%=
        request.getAttribute("ubal") %></h3> <h3>You are a Regular
        customer</h3>
</body></html>
```

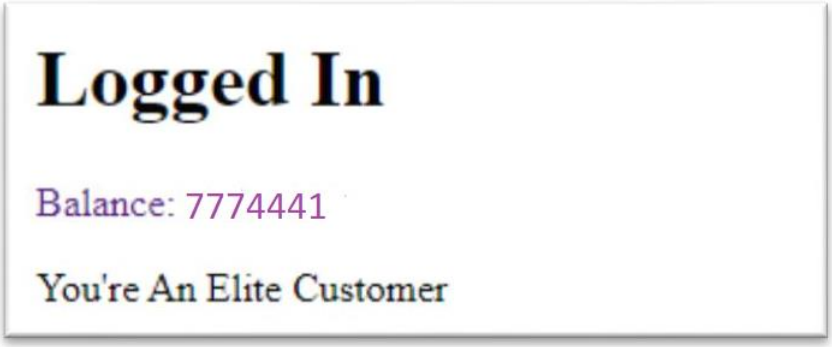
Output :



Enter Details

Krushnaraj

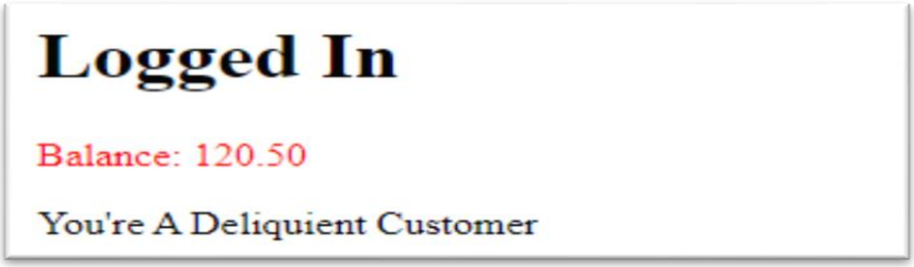
••••••••••



Logged In

Balance: 7774441

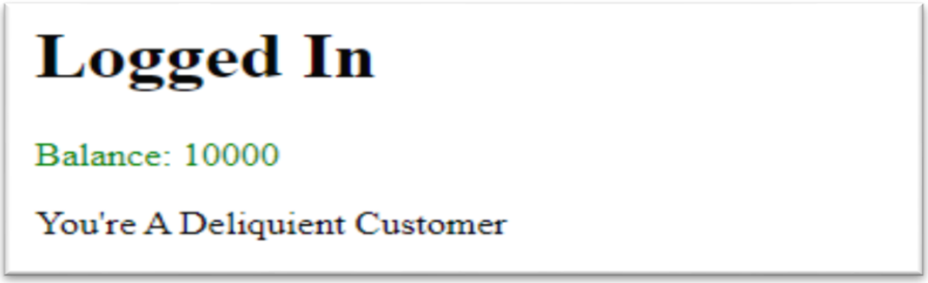
You're An Elite Customer



Logged In

Balance: 120.50

You're A Deliquient Customer



Logged In

Balance: 10000

You're A Deliquient Customer

Practical 16:

Develop a web application for an IT company's Job Application Portal. Develop profiles for admin and candidates applying for the job. Admin can view list of applications. Provide sign up and login feature for the candidates. They can register for one or more job profiles available. Make use of JSTL and verify if they are suitable for the job description and send confirmation message of the interview scheduled, else send rejection message. Access database using SQL tags.

Source Code (SignUp.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
	pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
	<form action="InsertSignUp.jsp" method="post">
		firstname:- <input type="text" name="fn"><br><br>
		middlename :- <input type="text" name="ln"><br><br>
		surname :- <input type="text" name="sn"><br><br>
		user-Id :- <input type="text" name="email"><br><br>
		MobileNo:- <input type="text" name="mobile"><br><br>
		password :- <input type="password" name="pass1"><br><br>
		<input type="hidden" name="key" value="SignUp"><input
			type="submit" value="SIGNUP">
	</form>
	<a href="Login.jsp">Already have account?</a>
</body></html>
```

Source Code (InsertSignUp.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ taglib uri="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:update var="result"dataSource="${dbsource}">
        insert into signup(firstname,middlename,surname,email,password,mobile) values
('${param.fn}','${param.ln}','${param.sn}','${param.email}','${param.pass1}','${param.mobile}'
);
</sql:update>
        <c:if test="${result ne null}">
        Account Created
<a href="Login.jsp">login</a>
        </c:if>
</body>
</html>
```

Source Code (Login.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title>
</head>
<body>
        <form action="VerifyLogin.jsp" method="post">
                USER ID:-<input type="text" name="uid"><br>
                <br>PassWord:-<input type="password" name="pw"><br>
                <br><input type="submit" value="login">
        </form>
</body>
</html>
```

Source Code (VerifyLogin.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head><title>Insert title here</title>
</head>
<body>
    <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
        url="jdbc:mysql://localhost/test" user="root" password="root" />
    <sql:query var="result"dataSource="${dbsource}">
    select * from signup where email='${param.uid}' and password='${param.pw}';
</sql:query>
    <c:choose>
        <c:when test="${result.rowCount != 0 }">
            <c:set scope="session" var="userid" value="${param.uid}"></c:set>
            <c:set scope="session" var="password" value="${param.pw}"></c:set>
            <c:if
                test="${param.uid eq 'admin@gmail.com' and param.pw eq
'admin'}">
                <c:redirecturl="WelcomeAdmin.jsp"></c:redirect>
            </c:if>
            <c:redirecturl="WelcomeUser.jsp"></c:redirect>
        </c:when>
        <c:otherwise>
            <c:redirecturl="Login.jsp"></c:redirect>
        </c:otherwise>
    </c:choose>
</body>
</html>
```

Source Code (WelcomeUser.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
    <h3>Submit a form</h3>
    <form action="InsertProfile.jsp" method="post">
```

```
                Full Name :- <input type="text" name="fullname"><br>
                <br> Email Address :- <input type="text" name="email"
                        value="${sessionScope.userid}"><br>
                <br> Address :- <input type="text" name="address"><br>
                <br> Mobile NO:-<input type="text" name="mobile"><br>
                <br> Area of Interest:-<br>
                <br><select name="jobs">
                        <option value="Performance Coach">Performance Coach</option>
                        <option value="Account Manager">Account Manager</option>
                        <option value="Administrative">Administrative</option>
                        <option value="HR">HR</option>
                        <option value="Sales">Sales</option>
                </select><br><br><input type="submit">
        </form>
        <a href="ViewStatus.jsp">Already Applied? View Status</a>
</body></html>
```

Source Code (InsertProfile.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title>
</head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:updatedataSource="${dbsource}" var="result">
        insert into jobprofile (email,fullname,address,mobile,job)
values('${param.email}','${param.fullname}','${param.address}','${param.mobile}','${param.job
s}');
</sql:update>
        <c:choose>
                <c:when test="${result.rowCount != 0 }">
                        <c:redirecturl="ViewStatus.jsp"></c:redirect>
                </c:when>
                <c:otherwise>
                        <c:redirecturl="WelcomeUser.jsp"></c:redirect>
                </c:otherwise>
        </c:choose>
</body></html>
```

Source Code (CheckStatus.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:query var="result"dataSource="${dbsource}">
        select accept,reject from jobprofile where email='${sessionScope.userid}';
</sql:query>
        <c:forEach var="i" items="${result.rows}">
                <c:choose>
                        <c:when test="${i.accept eq true }">accepted</c:when>
                        <c:when test="${i.reject eq true }">rejected</c:when>
                        <c:otherwise>pending</c:otherwise>
                </c:choose>
        </c:forEach>
</body>
</html>
```

Source Code (WelcomeAdmin.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:query var="result"dataSource="${dbsource}">
 select * from jobprofile where accept=false and reject=false;
</sql:query>
        <table border="1px solid black">
                <tr>
                        <th>fullname</th>
                        <th>Email</th>
                        <th>address</th>
                        <th>mobile</th>
                        <th>job</th>
                        <th>Action</th>
                </tr>
                <c:forEach items="${result.rows}" var="i">
                        <tr>
                                <td>${i.fullname}</td>
                                <td>${i.email}</td>
                                <td>${i.address}</td>
                                <td>${i.mobile}</td>
                                <td>${i.job}</td>
                                <td><a href="Accept.jsp?id=${i.id}"><input type="button"
                                                value="accept"></a></td>
                                <td><a href="Reject.jsp?id=${i.id}"><input type="button"
                                                value="reject"></a></td>
                        </tr>
                </c:forEach>
        </table>
</body>
</html>
```

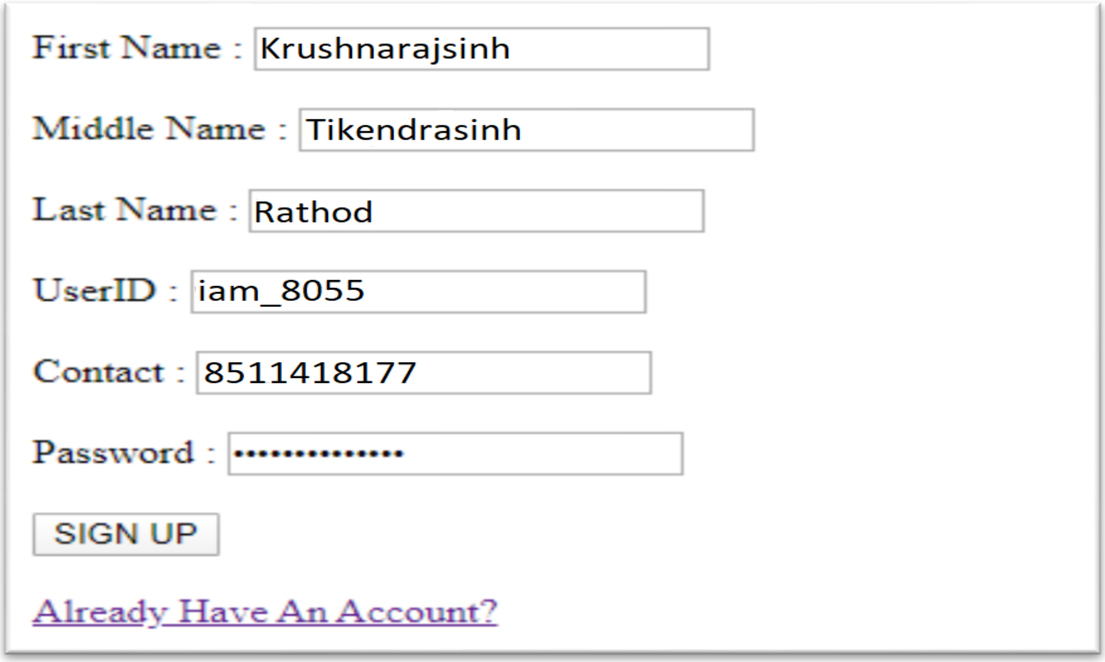
Source Code (Accept.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:update var="result"dataSource="${dbsource}">
        update jobprofile set accept=true where id='${param.id}';
</sql:update>
        <c:redirecturl="WelcomeAdmin.jsp"></c:redirect>
</body>
</html>
```

Source Code (Reject.jsp) :

```
<%@ page language="java"contentType="text/html; charset=ISO-8859-1"
        pageEncoding="ISO-8859-1"%>
<%@ tagliburi="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
<%@ tagliburi="http://java.sun.com/jstl/core_rt" prefix="c"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
        <sql:setDataSource var="dbsource" driver="com.mysql.jdbc.Driver"
                url="jdbc:mysql://localhost/test" user="root" password="root" />
        <sql:update var="result"dataSource="${dbsource}">
        update jobprofile set reject=true where id='${param.id}';
</sql:update>
        <c:redirecturl="WelcomeAdmin.jsp"></c:redirect>
</body></html>
```

Output :



First Name : Krushnarajsinh

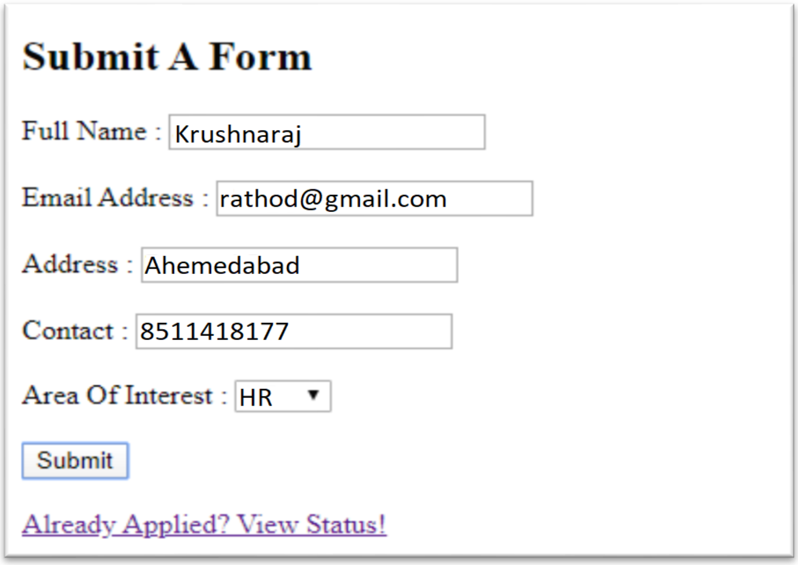
Middle Name : Tikendrasinh

Last Name : Rathod

UserID : iam_8055

Contact : 8511418177

Password : •••••••••••••••



Submit A Form

Full Name : Krushnaraj

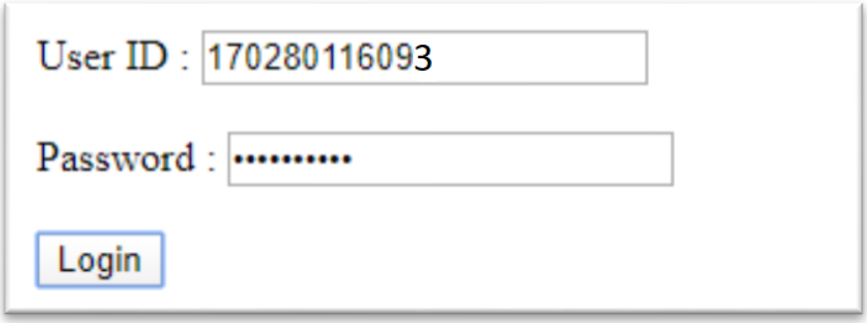
Email Address : rathod@gmail.com

Address : Ahemedabad

Contact : 8511418177

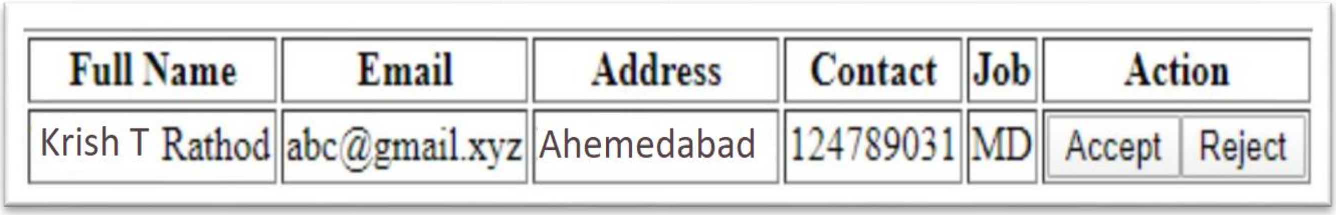
Area Of Interest : HR ▾

Already Applied? View Status!

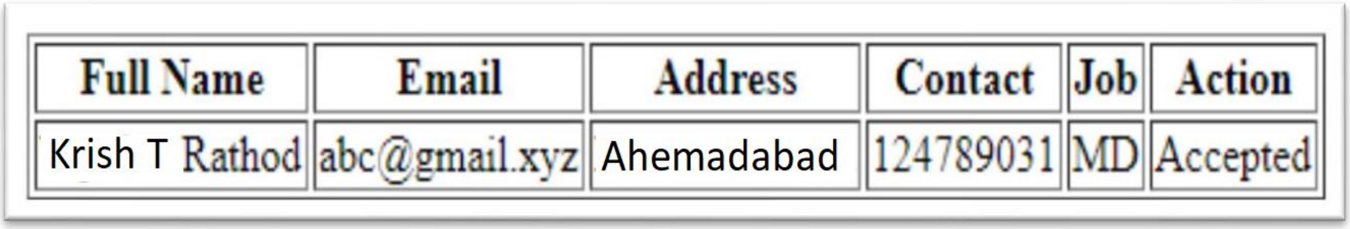


User ID : 170280116093

Password : •••••••••••



Full Name	Email	Address	Contact	Job	Action
Krish T Rathod	abc@gmail.xyz	Ahemedabad	124789031	MD	Accept Reject



Full Name	Email	Address	Contact	Job	Action
Krish T Rathod	abc@gmail.xyz	Ahemadabad	124789031	MD	Accepted

Practical 17:

Develop a web application that converts entered value from one currency (C1) to another (C2) on button click. Make use of JSF. Provide various currency options for both currencies, C1 as well as C2 using drop down menu.

Source Code (First.xhtml) :

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:h="http://xmlns.jcp.org/jsf/html"
    xmlns:f="http://xmlns.jcp.org/jsf/core"
xmlns:ui="http://xmlns.jcp.org/jsf/facelets">
    <h:head>
    <title>hello world</title>
    </h:head>
    <h:body>
            <h:form>
                    select Input type of Currency(c1)<br/><br/>
                    <h:selectOneMenu value="#{common.c1}">
                            <f:selectItemitemLabel="INR"itemValue="INR"></f:selectItem>
                            <f:selectItemitemLabel="USD"itemValue="USD"></f:selectItem>
                            <f:selectItemitemLabel="EURO"itemValue="EURO"></f:selectItem>
                            <f:selectItemitemLabel="DINAR"itemValue="DINAR"></f:selectItem>
                    </h:selectOneMenu><br/><br/>
                    Enter the amount <h:inputText id="currency" value="#{common.currency}">
                    <f:converterconverterId="javax.faces.Double"></f:converter>
                    </h:inputText><br/><br/>
                    select output type of Currency(c2)<br/><br/>
                            <h:selectOneMenu value="#{common.c2}">
                            <f:selectItemitemLabel="INR"itemValue="INR"></f:selectItem>
                            <f:selectItemitemLabel="USD"itemValue="USD"></f:selectItem>
                            <f:selectItemitemLabel="EURO"itemValue="EURO"></f:selectItem>
                            <f:selectItemitemLabel="DINAR"itemValue="DINAR"></f:selectItem>
                    </h:selectOneMenu><br/><br/>
                    <h:commandButton value="submit" action="Second"></h:commandButton>
            </h:form>
    </h:body>
</html>
```

Source Code (Second.xhtml) :

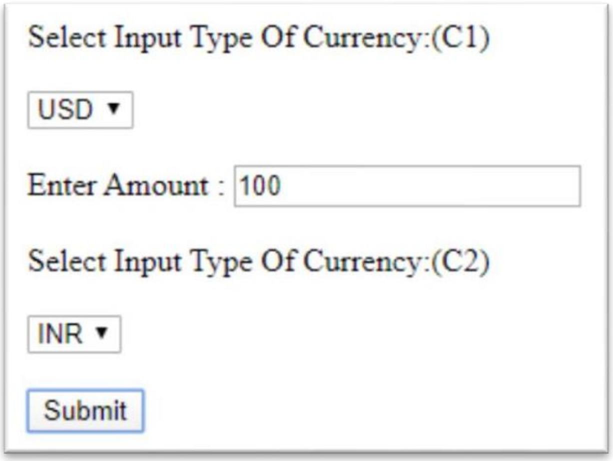
```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:h="http://xmlns.jcp.org/jsf/html"
    xmlns:ui="http://xmlns.jcp.org/jsf/facelets">
<h:head>
    <title>hello world</title>
</h:head>
<h:body>
            #{common.currency} #{common.c1} = #{common.outputcurrency} #{common.c2}
    </h:body>
</html>
```

Source Code (Common.java) : Managed Beans

```
import javax.faces.bean.ManagedBean;
@ManagedBean
public class Common {
    private String currency;
    private String c1;
    private String c2;
    private String outputcurrency;
    public String getOutputcurrency() {
            double output = calculate(c1, c2, currency);
            System.out.println(output);
            outputcurrency = String.valueOf(output);
            return outputcurrency;
    }
    public void setOutputcurrency(String outputcurrency) {
            this.outputcurrency = outputcurrency;
    }
    public String getC2() {
            return c2;
    }
    public void setC2(String c2) {
            this.c2 = c2;
    }
    private double calculate(String c12, String c22, String currency2) {
            // TODO Auto-generated method stub
            double input = Double.parseDouble(currency2);
            double output;
            System.out.println(c12 + "" + c22 + "" + currency2);
            if (c12.equalsIgnoreCase("USD")) {
                    input = input * 75.03;

            } else if (c12.equalsIgnoreCase("EURO")) {
                    input = input * 81.54;
            } else if (c12.equalsIgnoreCase("DINAR")) {
                    input = input * 242.05;
            } else {
            }
            if (c22.equalsIgnoreCase("USD")) {
                    output = input / 75.03;
            } else if (c22.equalsIgnoreCase("EURO")) {
                    output = input / 81.54;
            } else if (c22.equalsIgnoreCase("DINAR")) {
                    output = input / 242.05;
            } else {
                    output = input;
            }
            return output;
    }

    public String getC1() {
            return c1;
    }
    public void setC1(String c1) {
            this.c1 = c1;
    }
    public String getCurrency() {
            return currency;
    }
    public void setCurrency(String currency) {
            this.currency = currency;
    }
}
```

Output :



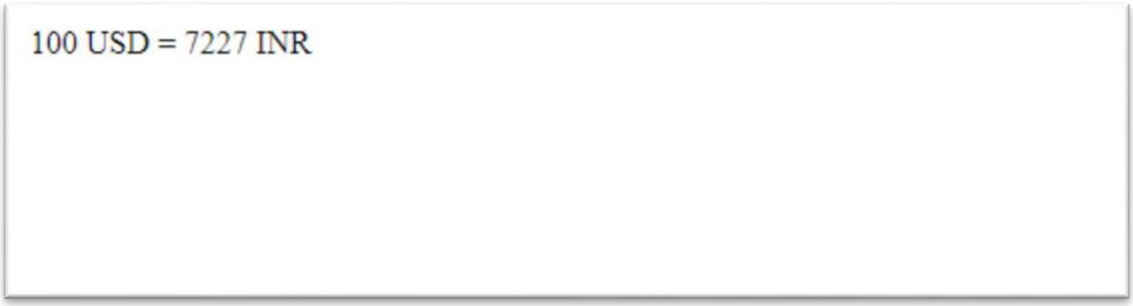
Select Input Type Of Currency:(C1)

USD ▾

Enter Amount : 100

Select Input Type Of Currency:(C2)

INR ▾



100 USD = 7227 INR

Practical 18:

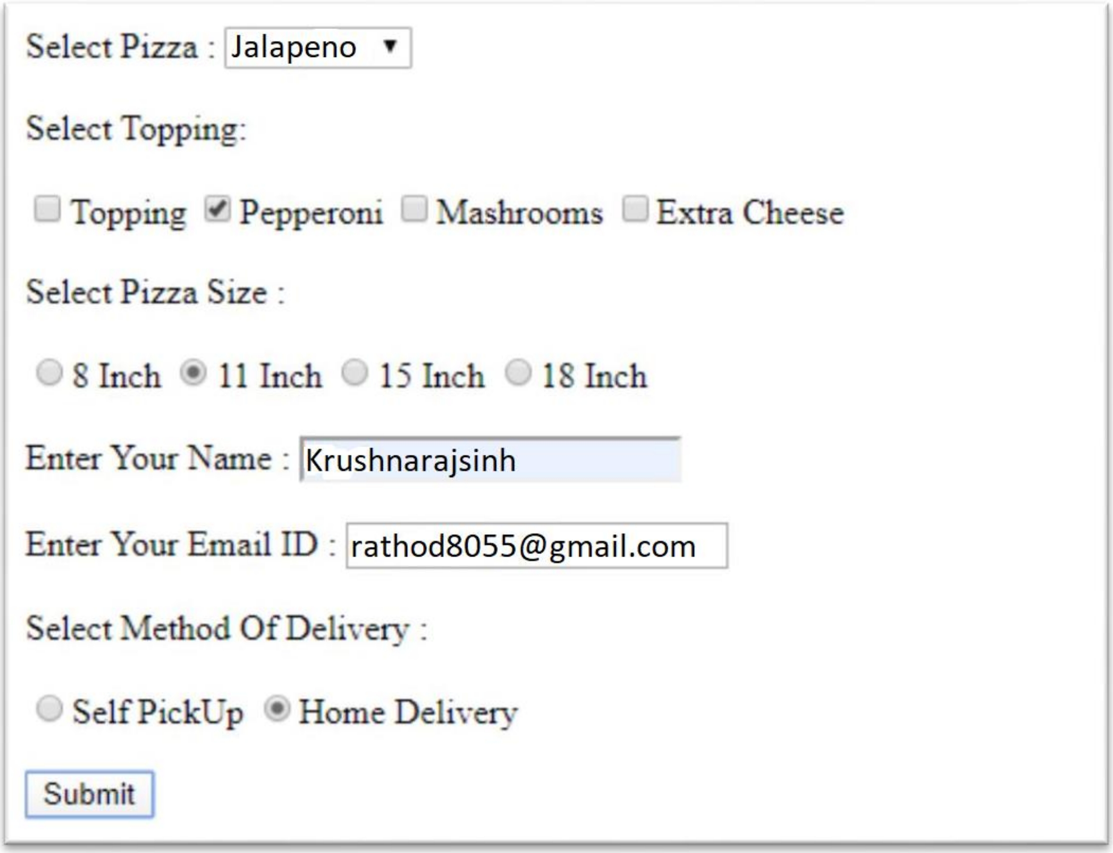
Use all facelet tags, convertor tags and validation tags in any of the practicals given or create your own.

Source Code : (index.html)

```
<?xml version="1.0" encoding="UTF-8"?>
<html xmlns="http://www.w3.org/1999/xhtml"
        xmlns:h="http://xmlns.jcp.org/jsf/html"
        xmlns:f="http://xmlns.jcp.org/jsf/core"
<h:head>
<style>
    .error{color:red;}
</style>
<title>Practical : 18</title>
</h:head>
<h:body>
<h:form id="/">
<h:outputLabel for="pizzaSelect" value="Select any pizza :" ></h:outputLabel>
<h:selectOneMenu id="pizzaSelect" value="#{pizza.name}"
requiredMessage="Please select any one pizza">
<f:selectItemitemValue="Margherita" itemLabel="Margherita"></f:selectItem>
<f:selectItemitemValue="Marinara" itemLabel="Marinara"></f:selectItem>
<f:selectItemitemValue="Carbonara" itemLabel="Carbonara"></f:selectItem>
<f:selectItemitemValue="7-Cheezy" itemLabel="7-Cheezy"></f:selectItem>
</h:selectOneMenu>
<br/><br/>
<h:outputLabel for="topping" value="Select any topping :" ></h:outputLabel>
<h:selectManyCheckbox id="topping" value="#{pizza.toppingName}" required="true">
<f:selectItemitemValue="Pepperoni" itemLabel="Pepperoni"></f:selectItem>
<f:selectItemitemValue="Mushrooms" itemLabel="Mushrooms"></f:selectItem>
<f:selectItemitemValue="Extra cheese" itemLabel="Extra cheese"></f:selectItem>
</h:selectManyCheckbox>
<br/>
<h:outputLabel for="size" value="Select any pizza size :" ></h:outputLabel>
<h:message for="size" styleClass="error"></h:message>
<h:selectOneRadio id="size" value="#{pizza.pizzaSize}" required="true"
requiredMessage="Please select size of pizza">
<f:selectItemitemValue="8 inch" itemLabel="8 inch"></f:selectItem>
<f:selectItemitemValue="11 inch" itemLabel="11 inch"></f:selectItem>
<f:selectItemitemValue="15 inch" itemLabel="15 inch"></f:selectItem>
<f:selectItemitemValue="18 inch" itemLabel="18 inch"></f:selectItem>
```

```
</h:selectOneRadio>
<br/>
<h:outputLabel for="UserName" value="Enter your name :" ></h:outputLabel>
<h:inputText id="UserName" value="#{pizza.userName}"
validatorMessage="length Of Username Should be greater than 3"
requiredMessage="username must be required" required="true">
<f:validateLength minimum="3"/>
</h:inputText>
<h:message for="UserName" styleClass="error"></h:message>
<br/><br/>
<h:outputLabel for="phone" value="Enter your phone number :" ></h:outputLabel>
<h:inputText id="phone" value="#{pizza.phoneNumber}"
      required="true" requiredMessage="phone number must be required"
validatorMessage="phone number length should be 10"
converterMessage="input should be numeric">
<f:converterconverterId="javax.faces.Long"></f:converter>
<f:validateLength minimum="10"/>
</h:inputText>
<h:message for="phone" styleClass="error"></h:message>
<br/><br/>
<h:outputLabel for="email" value="Enter your email :" ></h:outputLabel>
<h:inputText id="email" value="#{pizza.email}"
      required="true" requiredMessage="email must be required"
validatorMessage="please enter email in valid format" >
<f:validateRegex pattern="^([a-zA-Z_]{1,}@[a-z]{3,}[.]{1}[a-zA-
Z]{2,7})$"></f:validateRegex>
</h:inputText>
<h:message for="email" styleClass="error"></h:message>
<br/><br/>
<h:outputLabel for="delivery" value="Select any delivery method :" ></h:outputLabel>
<h:message for="delivery" styleClass="error"></h:message>
<h:selectOneRadio id="delivery" value="#{pizza.deliveryMethod}" required="true"
requiredMessage="please select any one method" >
<f:selectItemitemValue="Self Pickup" itemLabel="Self Pickup"></f:selectItem>
<f:selectItemitemValue="Home Delivery" itemLabel="Home Delivery"></f:selectItem>
</h:selectOneRadio>
<br/>
<h:commandButton value="Order It" action="#{pizza.calculatePrize}"></h:commandButton>
</h:form>
</h:body>
</html>
}
```

Output :



Select Pizza : Jalapeno ▼

Select Topping:

□ Topping ☑ Pepperoni □ Mashrooms □ Extra Cheese

Select Pizza Size :

○ 8 Inch ● 11 Inch ○ 15 Inch ○ 18 Inch

Enter Your Name : Krushnarajsinh

Enter Your Email ID : rathod8055@gmail.com

Select Method Of Delivery :

○ Self PickUp ● Home Delivery

Practical 19:

Create a Pizza Menu and price calculator using JSF. The input form should include:

- **Dropdown menu for choice of pizza bottom**
- **8-10 checkboxes for choice of pizza toppings**
- **Radio buttons to select pizza size**
- **Text fields for user to enter name and phone number**
- **Radio button to choose delivery method.**

When you click on the submit button you come to the output file where all of one's choices are listed, how much they cost and what the total cost will be.

Source Code : (index.html)

```
<?xml version="1.0" encoding="UTF-8"?>
<html xmlns="http://www.w3.org/1999/xhtml"
          xmlns:h="http://xmlns.jcp.org/jsf/html"
          xmlns:f="http://xmlns.jcp.org/jsf/core">
<h:head>
<style>
    .error{
color:red;
     }
</style>
<title>Practical : 19</title>
</h:head>
<h:body>
<h:form id="pizzaMenu">
<h:outputLabel for="pizzaSelect" value="Select any pizza :" ></h:outputLabel>
<h:selectOneMenu id="pizzaSelect" value="#{pizza.name}"
requiredMessage="Please select any one pizza">
<f:selectItemitemValue="Margherita" itemLabel="Margherita"></f:selectItem>
<f:selectItemitemValue="Marinara" itemLabel="Marinara"></f:selectItem>
<f:selectItemitemValue="Carbonara" itemLabel="Carbonara"></f:selectItem>
<f:selectItemitemValue="7-Cheezy" itemLabel="7-Cheezy"></f:selectItem>
</h:selectOneMenu>
```

```
<br/><br/>
<h:outputLabel for="topping" value="Select any topping :" ></h:outputLabel>
<h:selectManyCheckbox id="topping" value="#{pizza.toppingName}" required="true">
<f:selectItemitemValue="Pepperoni" itemLabel="Pepperoni"></f:selectItem>
<f:selectItemitemValue="Mushrooms" itemLabel="Mushrooms"></f:selectItem>
<f:selectItemitemValue="Extra cheese" itemLabel="Extra cheese"></f:selectItem>
</h:selectManyCheckbox>
<br/>
<h:outputLabel for="size" value="Select any pizza size :" ></h:outputLabel>
<h:message for="size" styleClass="error"></h:message>
<h:selectOneRadio id="size" value="#{pizza.pizzaSize}" required="true"
requiredMessage="Please select size of pizza">
<f:selectItemitemValue="8 inch" itemLabel="8 inch"></f:selectItem>
<f:selectItemitemValue="11 inch" itemLabel="11 inch"></f:selectItem>
<f:selectItemitemValue="15 inch" itemLabel="15 inch"></f:selectItem>
<f:selectItemitemValue="18 inch" itemLabel="18 inch"></f:selectItem>
</h:selectOneRadio>
<br/>
<h:outputLabel for="UserName" value="Enter your name :" ></h:outputLabel>
<h:inputText id="UserName" value="#{pizza.userName}"
validatorMessage="length Of Username Should be greater than 3"
requiredMessage="username must be required" required="true">
<f:validateLength minimum="3"/>
</h:inputText>
<h:message for="UserName" styleClass="error"></h:message>
<br/><br/>
<h:outputLabel for="phone" value="Enter your phone number :" ></h:outputLabel>
<h:inputText id="phone" value="#{pizza.phoneNumber}"
    required="true" requiredMessage="phone number must be required"
validatorMessage="phone number length should be 10"
converterMessage="input should be numeric">
<f:converterconverterId="javax.faces.Long"></f:converter>
<f:validateLength minimum="10"/>
</h:inputText>
<h:message for="phone" styleClass="error"></h:message>
<br/><br/>
<h:outputLabel for="email" value="Enter your email :" ></h:outputLabel>
<h:inputText id="email" value="#{pizza.email}"
    required="true" requiredMessage="email must be required"
validatorMessage="please enter email in valid format" >
<f:validateRegex pattern="^([a-zA-Z_]{1,}@[a-z]{3,}[.]{1}[a-zA-
Z]{2,7})$"></f:validateRegex>
</h:inputText>
<h:message for="email" styleClass="error"></h:message>
<br/><br/>
```

```
<h:outputLabel for="delivery" value="Select any delivery method :" ></h:outputLabel>
<h:message for="delivery" styleClass="error"></h:message>
<h:selectOneRadio id="delivery" value="#{pizza.deliveryMethod}" required="true"
requiredMessage="please select any one method" >
<f:selectItemitemValue="Self Pickup" itemLabel="Self Pickup"></f:selectItem>
<f:selectItemitemValue="Home Delivery" itemLabel="Home Delivery"></f:selectItem>
</h:selectOneRadio>
<br/>
<h:commandButton value="Order It" action="#{pizza.calculatePrize}"></h:commandButton>
</h:form>
</h:body>
</html>
```

Source Code : (PrintBIll.html)

```
<?xml version="1.0" encoding="UTF-8"?>
<html xmlns="http://www.w3.org/1999/xhtml"
        xmlns:h="http://xmlns.jcp.org/jsf/html"
        xmlns:f="http://xmlns.jcp.org/jsf/core"
        xmlns:c="http://xmlns.jcp.org/jsf/facelets">
<h:head>
        <title>Insert title here</title>
</h:head>
<h:body>
Dear #{pizza.userName} ,<br/>
<hr width="250px" style="float:left"/><br/>
Added items<br/>
<hr width="250px" style="float:left"/><br/>
Pizza : #{pizza.name} <br/><br/>
Pizza Size : #{pizza.pizzaSize} <br/><br/>
Toppinags : <c:repeat var="i" value="#{pizza.toppingName}">${i},</c:repeat><br/>
<hr width="250px" style="float:left"/><br/>
<hr width="250px" style="float:left"/><br/>
------------------------Payment--------------------<br/>
<hr width="250px" style="float:left"/><br/>
Pizza Price : #{pizza.pizzaPrice}<br/><br/>
Additional charges : <br/><br/>
Topping Price : #{pizza.toppingPrice}<br/><br/>
Delivery Charge : #{pizza.deliveryCharge}<br/>
<hr width="250px" style="float:left"/><br/>
Total Price : #{pizza.totalPrice}<br/>
<hr width="250px" style="float:left"/><br/>
</h:body>
</html>
```

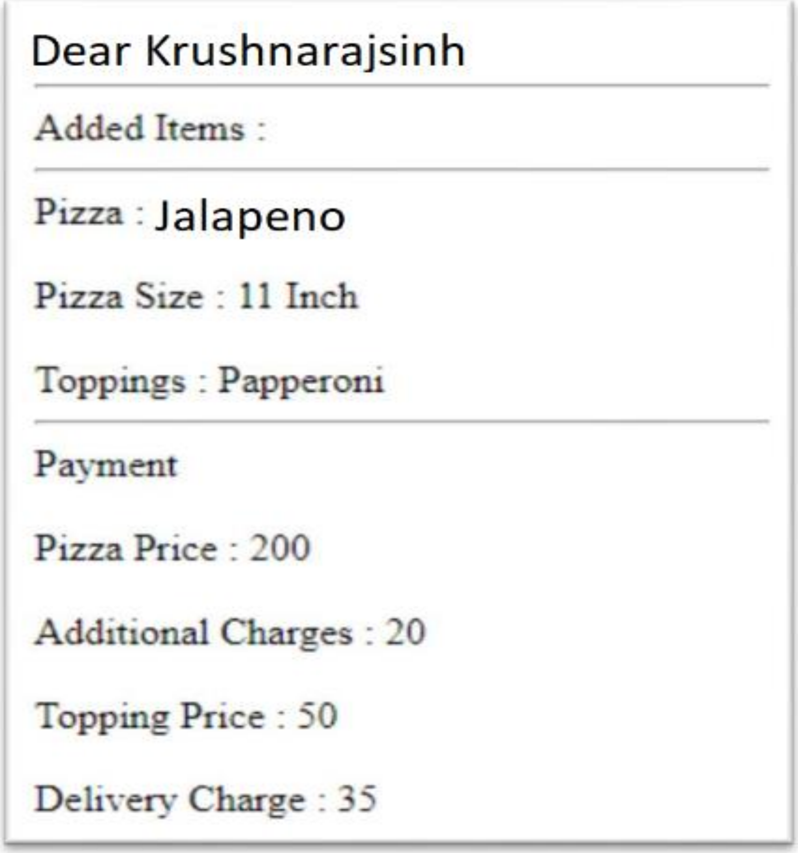
Source Code : (Menu.html)

```
import javax.faces.bean.ManagedBean;
@ManagedBean(name="pizza")
public class PizzaMenu {
        private String name;
        private String[] toppingName=new String[3];
        private String pizzaSize;
        private String userName;
        private Long phoneNumber;
        private String email;
        private String deliveryMethod;
        private double totalPrice;
        private double pizzaPrice;
        private double toppingPrice=0;
  private double deliveryCharge;
   private double pizzaSizePrice;
        public double getTotalPrice() {
                return totalPrice;
        }
        public double getPizzaPrice() {
                return pizzaPrice;

        public double getToppingPrice() {
                return toppingPrice;
        }
        public double getDeliveryCharge() {
                return deliveryCharge;
        }
        public double getPizzaSizePrice() {
                return pizzaSizePrice;
        }
        public String getEmail() {
                return email;
        }
        public void setEmail(String email) {
                this.email = email;
        }
        public void setName(String name) {
                this.name = name;
        }
        public String getName() {
                return name;
        }
```

```
public String[] getToppingName() {
        return toppingName;
}
public void setToppingName(String[] toppingName) {
        this.toppingName = toppingName;
}
public String getPizzaSize() {
        return pizzaSize;
}
public void setPizzaSize(String pizzaSize) {
        this.pizzaSize = pizzaSize;
}
public String getUserName() {
        return userName;
}
public void setUserName(String userName) {
        this.userName = userName;
}
public Long getPhoneNumber() {
        return phoneNumber;
}
public void setPhoneNumber(Long phoneNumber) {
        this.phoneNumber = phoneNumber;
}
public String getDeliveryMethod() {
        return deliveryMethod;
}
public void setDeliveryMethod(String deliveryMethod) {
        this.deliveryMethod = deliveryMethod;
}
public String calculatePrize()
{
        if(name.equals("Margherita"))
                this.pizzaPrice=200;
        else if(name.equals("Marinara"))
                this.pizzaPrice=250;
        else if(name.equals("Carbonara"))
                this.pizzaPrice=300;
        else if(name.equals("7-Cheezy"))
                this.pizzaPrice=400;
        for(int i=0;i<toppingName.length;i++)
        {
        if(toppingName[i].equals("Pepperoni"))
                this.toppingPrice+=20;
        else if(toppingName[i].equals("Mushrooms"))
```

```
                this.toppingPrice+=50;
        else if(toppingName[i].equals("Extra cheese"))
                this.toppingPrice+=60;
        }
        if(pizzaSize.equals("8 inch"))
                this.pizzaSizePrice=0;
        else if(pizzaSize.equals("11 inch"))
                this.pizzaSizePrice=50;
        else if(pizzaSize.equals("15 inch"))
                this.pizzaSizePrice=200;
        else if(pizzaSize.equals("18 inch"))
                this.pizzaSizePrice=400;
        if(deliveryMethod.equals("Self Pickup"))
                this.deliveryCharge=0;
        else if(deliveryMethod.equals("Home Delivery"))
                this.deliveryCharge=30
        this.totalPrice=pizzaPrice+toppingPrice+pizzaSizePrice+deliveryCharge
        return "printBill";
    }   }
```

Output :



```
Dear Krushnarajsinh

Added Items :

Pizza : Jalapeno

Pizza Size : 11 Inch

Toppings : Papperoni

Payment

Pizza Price : 200

Additional Charges : 20

Topping Price : 50

Delivery Charge : 35
```

Practical 20:

Implement Student Information System (enrolment no., name, semester, SPI, CPI, backlogs, email, contact no., etc) using Hibernate. It should perform operations such as inserting student record, searching the records, displaying all records, editing the record, deleting record, displaying top performer(s), etc.

Source Code : (AddStudent.java)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
<form action="<%=request.getContextPath() %>/StudentController" method="post">
<table border="1px solid black">
<tr>
        <td>Enrollment</td>
        <td><input type="text" name="en"></td>
</tr>
<tr>
        <td>NAME</td>
        <td><input type="text" name="name"></td>
</tr>
<tr>
        <td>SEMESTER</td>
        <td><input type="text" name="sem"></td>
</tr>
<tr>
        <td>BACKLOG</td>
        <td><input type="text" name="backlog"></td>
</tr>
<tr>
        <td>EMAIL</td>
        <td><input type="text" name="email"></td>
</tr>
<tr>
        <td>CONTACT NO</td>
        <td><input type="text" name="contact"></td>
</tr>
<tr>
        <td>SPI</td>
        <td><input type="text" name="spi"></td>
</tr>
```

```
<tr>
        <td>CPI</td>
        <td><input type="text" name="cpi"></td>
</tr>

<tr>
        <td colspan="2">
                <center><input type="submit" value="Register">
                <input type="hidden" name="key" value="insert">
                <a href="StudentController?key=search"><input type="button"
                value="Search"></a></center>

        </td>
</tr>
</table>
</form>
</body>
</html>
```

Source Code : (EditStudentDetail.java)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@taglib prefix="c" uri="http://java.sun.com/jstl/core_rt" %>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
<form Action="<%=request.getContextPath() %>/StudentController" method="post">
<c:forEach items="${sessionScope.listEdit}" var="object">
<input type="text" name="name" value="${object.name}">
<input type="text" name="sem" value="${object.sem}">
<input type="text" name="spi" value="${object.spi}">
<input type="text" name="cpi" value="${object.cpi}">
<input type="text" name="email" value="${object.email}">
<input type="text" name="contact" value="${object.contact}">
<input type="text" name="backlog" value="${object.backlog}">
<input type="text" name="en" value="${object.enrollment}">
<input type="hidden" name="id" value="${object.id}"/>
<input type="hidden" name="key" value="update"/><br><br>
<input type="submit" value="Update">
</c:forEach>
</form>
</body>
</html>
```

Source Code : (ViewStudentDetails.java)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
```

```
<%@taglib prefix="c" uri="http://java.sun.com/jstl/core_rt" %>
<!DOCTYPE html>
<html>
<head><title>Insert title here</title></head>
<body>
<table border="1" style="border-collapse: collapse;">
<tr>
<td>Enrollment</td>
<td>Name</td>
<td>Semester</td>
<td>SPI</td>
<td>CPI</td>
<td>Backlog</td>
<td>Email</td>
<td>Contact NO</td>
<td>Delete</td>
<td>Edit</td>
</tr>
<c:forEach items="${sessionScope.listVO}" var="object">
<tr>
<td>${object.enrollment}</td>
<td>${object.name}</td>
<td>${object.sem}</td>
<td>${object.spi}</td>
<td>${object.cpi}</td>
<td>${object.backlog}</td>
<td>${object.email}</td>
<td>${object.contact}</td>
<td><a href="<%=request.getContextPath()
%>/StudentController?key=delete&id=${object.id}">Delete</a></td>
<td><a href="<%=request.getContextPath()
%>/StudentController?key=edit&id=${object.id}">Edit</a></td>
</tr>
</c:forEach>
</table>
</body>
</html>
```

Source Code : (StudentController.java)

```
import java.io.IOException;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;

@WebServlet("/StudentController")
public class StudentController extends HttpServlet {
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
```

```
 ServletException, IOException {
  String key=request.getParameter("key");
        If(key.equals("search")){
                         search(request,response);
 }else if(key.equals("edit")){
                         editSearch(request,response);
 }else if(key.equals("delete")){
                         delete(request,response);
 }
}
private void editSearch(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException{
                             int id=Integer.parseInt(request.getParameter("id"));
                            StudentDAOdao=new StudentDAO();
                            List<StudentVO>listEdit=dao.editSearch(id);
                         HttpSession session=request.getSession();
                         session.setAttribute("listEdit",listEdit);
                         response.sendRedirect("editStudentDetail.jsp");
 }
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
 ServletException, IOException {
                         String key=request.getParameter("key");
                         if(key.equals("insert")){
                         insert(request,response);
                         }else if(key.equals("update")){
                             update(request,response);
                         }
 }
    private void delete(HttpServletRequest request, HttpServletResponse response)throws
 ServletException, IOException {
                         int id=Integer.parseInt(request.getParameter("id"));
                         StudentVOvo=new StudentVO();
                         vo.setId(id);
 StudentDAOdao=new StudentDAO();
                         dao.delete(vo);
                         search(request, response);
 }
 private void update(HttpServletRequest request, HttpServletResponse response)throws
 ServletException, IOException {
                         int id=Integer.parseInt(request.getParameter("id"));
                         String enrollment=request.getParameter("en");
 String name=request.getParameter("name");
                         int sem=Integer.parseInt(request.getParameter("sem"));
                         int backlog=Integer.parseInt(request.getParameter("backlog"));
                         String email=request.getParameter("email");
```

```
                              String contact=request.getParameter("contact");
                              double spi=Double.parseDouble(request.getParameter("spi"));
                              double cpi=Double.parseDouble(request.getParameter("cpi"));
                              StudentVOvo=new StudentVO();
                                  vo.setId(id);
                                  vo.setEnrollment(enrollment);
vo.setName(name);
                                  vo.setSem(sem);
                                  vo.setBacklog(backlog);
                                  vo.setEmail(email);
                                  vo.setContact(contact);
                                  vo.setSpi(spi);
                                  vo.setCpi(cpi);
                                  StudentDAOdao=new StudentDAO();
                                  dao.update(vo);
                                  search(request, response);
   }
private void search(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
StudentDAOdao=new StudentDAO();
     List<StudentVO>listVO=dao.search();
HttpSession session=request.getSession();
session.setAttribute("listVO",listVO);
response.sendRedirect("viewStudentDetail.jpg");
}
private void insert(HttpServletRequest request, HttpServletResponse response)throws
ServletException, IOException {
                                  String enrollment=request.getParameter("en");
                                  String name=request.getParameter("name");
                                  int sem=Integer.parseInt(request.getParameter("sem"));
                                  int backlog=Integer.parseInt(request.getParameter("backlog"));
                                String email=request.getParameter("email");
                                  String contact=request.getParameter("contact");
                                  double spi=Double.parseDouble(request.getParameter("spi"));
                                  double cpi=Double.parseDouble(request.getParameter("cpi"));

                                  StudentVOvo=new StudentVO();
                                  vo.setEnrollment(enrollment);
                                  vo.setName(name);
                                  vo.setSem(sem);
                                  vo.setBacklog(backlog);
                                  vo.setEmail(email);
                                  vo.setContact(contact);
                                  vo.setSpi(spi);
                                  vo.setCpi(cpi);
```

```
                                        StudentDAOdao=new StudentDAO();
                                        dao.insert(vo);
                                        search(request,response);

}

}
```

Source Code : (StudentVO.java)

```
public class StudentVO {
private int id;
private String enrollment,name,email,contact;
private int sem,backlog;
private double spi,cpi;
public int getId() {
        return id;
}
public void setId(int id) {
        this.id = id;
}
public String getEnrollment() {
        return enrollment;
}
public void setEnrollment(String enrollment) {
        this.enrollment = enrollment;
}
public String getName() {
        return name;
}
public void setName(String name) {
        this.name = name;
}
public String getEmail() {
        return email;
}
public void setEmail(String email) {
        this.email = email;
}
public String getContact() {
        return contact;
}
public void setContact(String contact) {
        this.contact = contact;
}
public int getSem() {
        return sem;
}
public void setSem(int sem) {
        this.sem = sem;
}
public int getBacklog() {
        return backlog;
}
public void setBacklog(int backlog) {
        this.backlog = backlog;
}
public double getSpi() {
        return spi;
}
public void setSpi(double spi) {
        this.spi = spi;
}
public double getCpi() {
        return cpi;
}
public void setCpi(double cpi) {
        this.cpi = cpi;
}
}
```

Source Code : (StudentVO.hbm.xml)

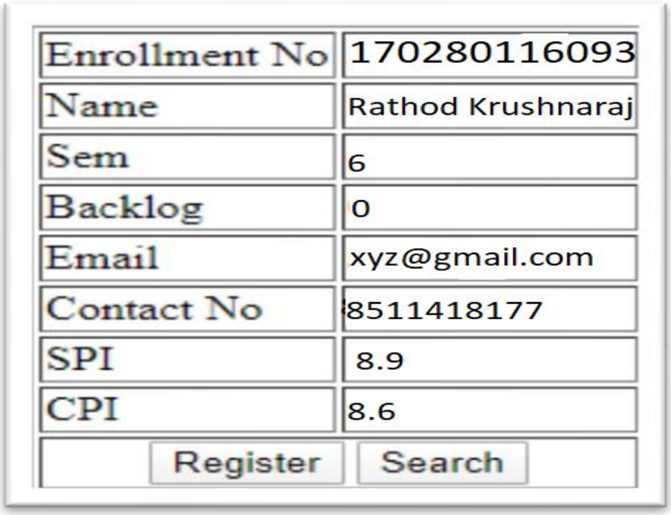
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
<class name="vo.StudentVO" table="Practicle19">
<id name="id" type="int" column="id">
<generator class="increment"></generator>
</id>
<property name="enrollment">
<column name="enrollment"></column>
</property>
<property name="name">
<column name="name"></column>
</property>
<property name="email">
<column name="email"></column>
</property>
<property name="contact">
<column name="contact"></column>
</property>
<property name="sem">
<column name="sem"></column>
</property>
```

```
  <property name="backlog">
  <column name="backlog"></column>
  </property>
  <property name="spi">
  <column name="spi"></column>
  </property>
  <property name="cpi">
  <column name="cpi"></column>
  </property>
  </class>
</hibernate-mapping>
```

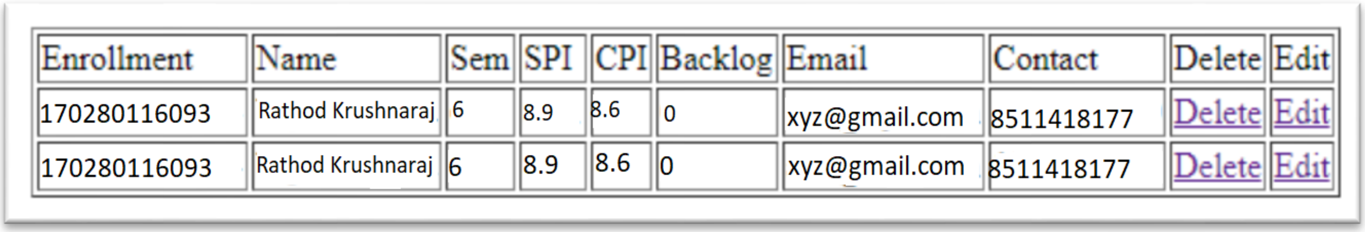
Source Code : (StudentDAO.java)

```
package dao;
import java.util.List;
import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;
import vo.StudentVO;
public class StudentDAO {
        public void insert(StudentVOvo) {
        SessionFactory sf=
new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
                Session session=sf.openSession();
                Transaction t=session.beginTransaction();
                session.save(vo);
                t.commit();
                session.close();
        }
        public List<StudentVO> search() {
                SessionFactory sf=
new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
Session session=sf.openSession();
                Query q=session.createQuery("from StudentVO");
                List<StudentVO> list=q.list();
                return list;
        }
        public List<StudentVO>editSearch(int id) {
                SessionFactory sf=
new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
                Session session=sf.openSession();
                Query q=session.createQuery("from StudentVO where id="+id);
                List<StudentVO> list=q.list();
                return list;
        }
        public void update(StudentVOvo) {
                SessionFactory sf=
new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
                Session session=sf.openSession();
                Transaction t=session.beginTransaction();
                session.update(vo);
                t.commit();
                session.close();
        }
        public void delete(StudentVOvo) {
                SessionFactory sf=
new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
Session session=sf.openSession();
                Transaction t=session.beginTransaction();
                session.delete(vo);
                t.commit();
                session.close();
        }  }
```

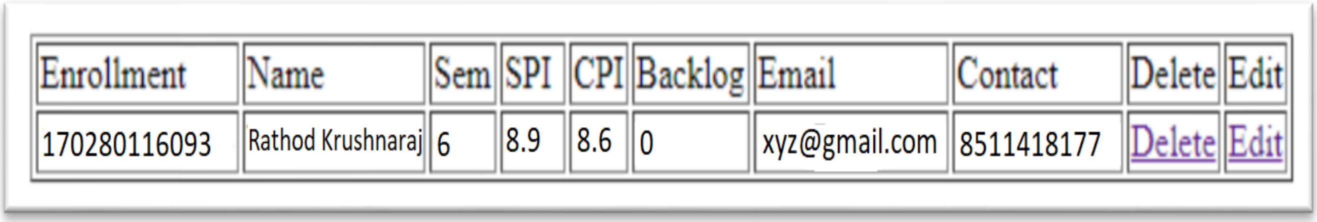
Output :



Enrollment No	170280116093
Name	Rathod Krushnaraj
Sem	6
Backlog	0
Email	xyz@gmail.com
Contact No	8511418177
SPI	8.9
CPI	8.6



Enrollment	Name	Sem	SPI	CPI	Backlog	Email	Contact	Delete	Edit
170280116093	Rathod Krushnaraj	6	8.9	8.6	0	xyz@gmail.com	8511418177	Delete	Edit
170280116093	Rathod Krushnaraj	6	8.9	8.6	0	xyz@gmail.com	8511418177	Delete	Edit



Enrollment	Name	Sem	SPI	CPI	Backlog	Email	Contact	Delete	Edit
170280116093	Rathod Krushnaraj	6	8.9	8.6	0	xyz@gmail.com	8511418177	Delete	Edit

Practical 21:

Implement Library Management System (book title, ISBN, author, rating, available copies, issued copies) using Spring Framework. It should perform operations such as adding book to library, issuing book(s) to a student, returning books, displaying all books, search operation, etc.

Source Code : (index.jsp)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="loadInsert.html">Insert Book</a><br><br>
<a href="displayAllBooks.html">Display All Books</a><br><br>
</body>
</html>
```

Source Code : (insertBook.jsp)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@taglib prefix="s" uri="http://www.springframework.org/tags/form" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

<s:form action="insertBook.html" method="post" modelAttribute="bookVO">

Book Title : <s:input path="book_title"/><br/><br/>
Author : <s:input path="author"/><br/><br/>
ISBN Number : <s:input path="isbn"/><br/><br/>
Total Copy : <s:input path="available_copy"/><br/><br/>
```

```
<input type="submit" value="Add Book">

</s:form>
</body>
</html>
```

Source Code : (ShowAll.jsp)

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@taglib prefix="c" uri="http://java.sun.com/jstl/core_rt" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
<script type="text/javascript">
 function search() {
                                   var isbn, filter, found, table, row, cell, i, j;
                                  isbn = document.getElementById("isbn");
                                     filter = isbn.value.toUpperCase();
                                     table = document.getElementById("table");
                                     row  = table.getElementsByTagName("tr");
                                     for (i = 1; i<row.length; i++) {
                                         cell = row[i].getElementsByTagName("td");
                                         for (j = 0; j <cell.length; j++) {
                                             if (cell[j].innerHTML.toUpperCase().indexOf(filter) > -1) {
                                                 found = true;
                                             }
                                         }
                                         if (found) {
                                             row[i].style.display = "";
                                             found = false;
                                         } else {
                                             row[i].style.display = "none";
                                         }
                                     }
}

</script>
</head>
<body>
<!-- <form action="searchOperation.html" method="post"> -->
Search By ISBN : <input type="text" id="isbn" onkeyup="search()"><!-- <input type="submit"
```

```
value="SEARCH"> -->
<!-- </form> -->
<table border="1" style="border-collapse: collapse;" id="table">
<tr>
<td>Book Title</td>
<td>Author</td>
<td>ISBN</td>
<td>Available Copy</td>
<td>Issued Copy</td>
<td>Action</td>
</tr>

<c:forEach items="${bookList}" var="object">
<tr>

<td>${object.book_title}</td>
<td>${object.author}</td>
<td>${object.isbn}</td>
<td>${object.available_copy}</td>
<td>${object.issued_copy}</td>
<td><a href="issue.html?isbn=${object.isbn}"><button>Issue</button></a><br>
<a href="returnBook.html?isbn=${object.isbn}"><button>Return Book</button></a></td>

</tr>
</c:forEach>
</table>
</body>
</html>
```

Source Code : (LibraryManage.java)

```
package com.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import com.dao.LibraryDAO;
import com.model.BookVO;
import com.sun.research.ws.wadl.Request;

@Controller
public class LibraryController {

@Autowired
private LibraryDAOlDao;

@RequestMapping(value="loadInsert.html", method=RequestMethod.GET)
   public ModelAndViewloadInsert()
   {
                                    System.out.println("hii");
                                    return new ModelAndView("insertBook","bookVO",new
BookVO());
   }

@RequestMapping(value="insertBook.html" , method=RequestMethod.POST)
public ModelAndViewinsertBook(@ModelAttributeBookVOvo)
{
lDao.insertBook(vo);
return new ModelAndView();
}

@RequestMapping(value=" displayAllBooks.html" , method=RequestMethod.GET)
public ModelAndViewdisplayAllBooks()
{
                                    List<BookVO> list=lDao.displayAllBooks();
                                    System.out.println(list.size());
                                    return new ModelAndView("displayAllBook","bookList",list);
}

@RequestMapping(value="issue.html" , method=RequestMethod.GET)
public ModelAndView issue(@RequestParam String isbn)
{
                                    BookVOvo=this.lDao.searchByISBN(isbn).get(0);
                                    vo.setIssued_copy(vo.getIssued_copy()+1);
                                    vo.setAvailable_copy(vo.getAvailable_copy()-1);
                                    this.lDao.issue(vo);
                                    List<BookVO> list=lDao.displayAllBooks();
                                    System.out.println(list.size());
                                    return new ModelAndView("displayAllBook","bookList",list);
}

@RequestMapping(value="returnBook.html" , method=RequestMethod.GET)
public ModelAndViewreturnBook(@RequestParam String isbn)
{
                                          BookVOvo=this.lDao.searchByISBN(isbn).get(0);
                                          vo.setAvailable_copy(vo.getAvailable_copy()+1);
                                          vo.setIssued_copy(vo.getIssued_copy()-1);
this.lDao.returnBook(vo);
                                          List<BookVO> list=lDao.displayAllBooks();
                                          System.out.println(list.size());
                                          return new ModelAndView("displayAllBook","bookList",list);
}

@RequestMapping(value="searchOperation.html" , method=RequestMethod.POST)
                                     public ModelAndViewsearchOperation(@RequestParam String isbn)
{
        List<BookVO> list=this.lDao.searchByISBN(isbn);
                                          System.out.println(list.size());
                                          return new ModelAndView("displayAllBook","bookList",list);
}
}
```

Source Code : (BookVO.java)

```
package com.model;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

@Entity
 @Table(name="BookDetails")
 public class BookVO {

                                     @Id
                                     @Column(name="id")
  @GeneratedValue(strategy=GenerationType.AUTO)
                                     private int id;

                                     @Column(name="book_title")
                                     private String book_title;
```

```
@Column(name="isbn")
private String isbn;

@Column(name="author")
private String author;

@Column(name="available_copy")
private int available_copy;

@Column(name="issued_copy")
private int issued_copy=0;

public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String getBook_title() {
    return book_title;
}

public void setBook_title(String book_title) {
    this.book_title = book_title;
}

public String getIsbn() {
    return isbn;
}

public void setIsbn(String isbn) {
    this.isbn = isbn;
}

public String getAuthor() {
    return author;
}

public void setAuthor(String author) {
    this.author = author;
}

public int getAvailable_copy() {
```

```
                                            return available_copy;
                                        }

                                        public void setAvailable_copy(int available_copy) {
                                            this.available_copy = available_copy;
                                        }

                                        public int getIssued_copy() {
                                            return issued_copy;
                                        }

                                        public void setIssued_copy(int issued_copy) {
                                            this.issued_copy = issued_copy;
                                        }
 }
```

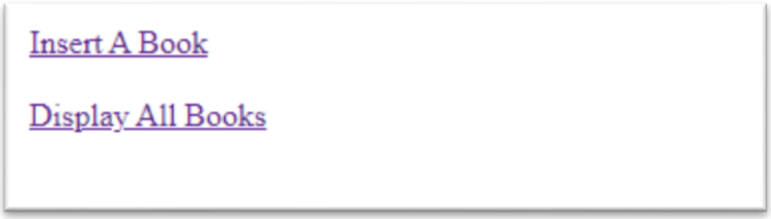
Source Code : (LibraryDAO.java)

```
 package com.dao;

import java.util.ArrayList;
 import java.util.List;
import org.hibernate.Query;
 import org.hibernate.Session;
 import org.hibernate.SessionFactory;
 import org.hibernate.Transaction;
 import org.springframework.beans.factory.annotation.Autowired;
 import org.springframework.stereotype.Repository;
import com.model.BookVO;

 @Repository
 public class LibraryDAO {

                                        @Autowired
                                        private SessionFactorysessionFactory;

                                        public void insertBook(BookVOvo)
                                        {
                                            try{
                                            Session session=sessionFactory.openSession();
                                            Transaction t=session.beginTransaction();
                                            session.save(vo);
                                            t.commit();
                                            }catch(Exception e){
                                                        System.out.println(e);
```

```
        }

    }

    public List<BookVO>displayAllBooks() {
        List<BookVO> list=new ArrayList<BookVO>();
        try {
        Session session=sessionFactory.openSession();
        Query q=session.createQuery("from BookVO");
        list=q.list();
        }catch (Exception e) {
        System.out.println(e);
        }
        return list;
    }

    public List<BookVO>searchByISBN(String isbn) {
        List<BookVO> list=new ArrayList<BookVO>();
        try {
        Session session=sessionFactory.openSession();
        Query q=session.createQuery("from BookVO where isbn like
'"+isbn+"%'");

        list=q.list();
        }catch (Exception e) {
        System.out.println(e);
        }
        return list;
    }

    public void issue(BookVO v) {
 try {

        Session session=sessionFactory.openSession();
        Transaction t=session.beginTransaction();
session.update(v);

        t.commit();
        session.close();
        } catch (Exception e) {
                System.out.println(e);
        }
    }

    public void returnBook(BookVOvo) {
        try {
        Session session=sessionFactory.openSession();
        Transaction t=session.beginTransaction();
```

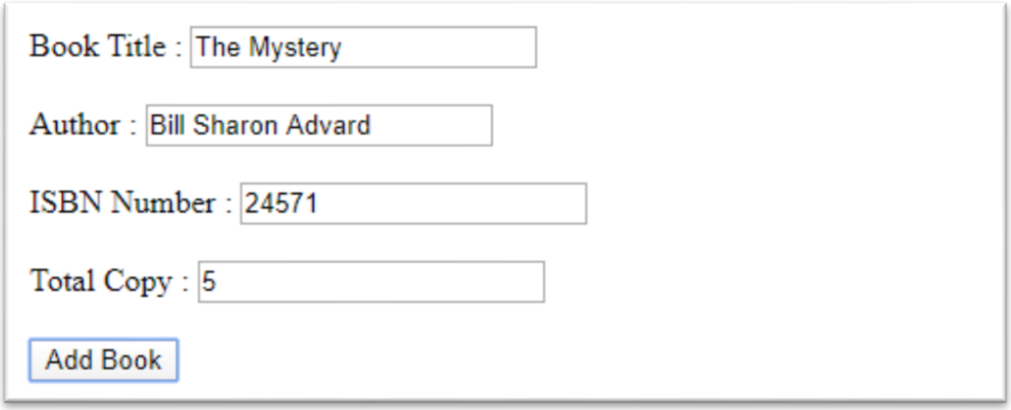
```
session.update(vo);
                                        t.commit();
                                        session.close();
                                        } catch (Exception e) {
                                                  System.out.println(e);
                                        }
                                    }

}
```

Output :



Insert A Book

Display All Books

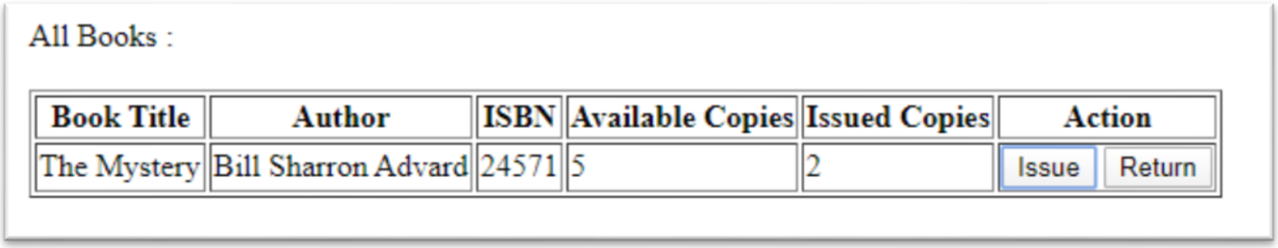


Book Title : The Mystery

Author : Bill Sharon Advard

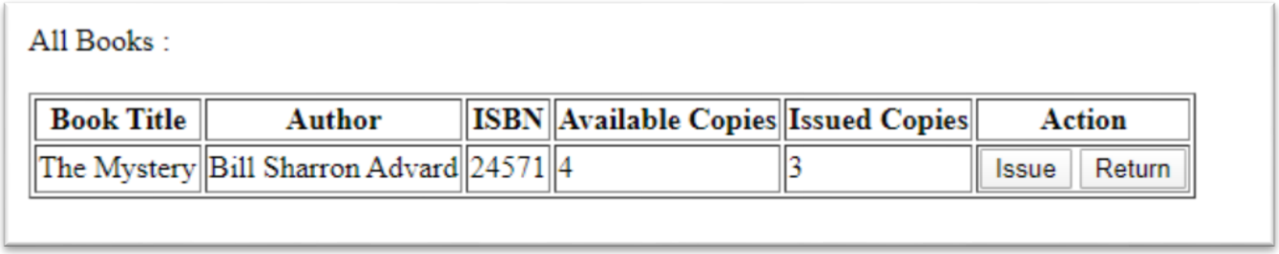
ISBN Number : 24571

Total Copy : 5



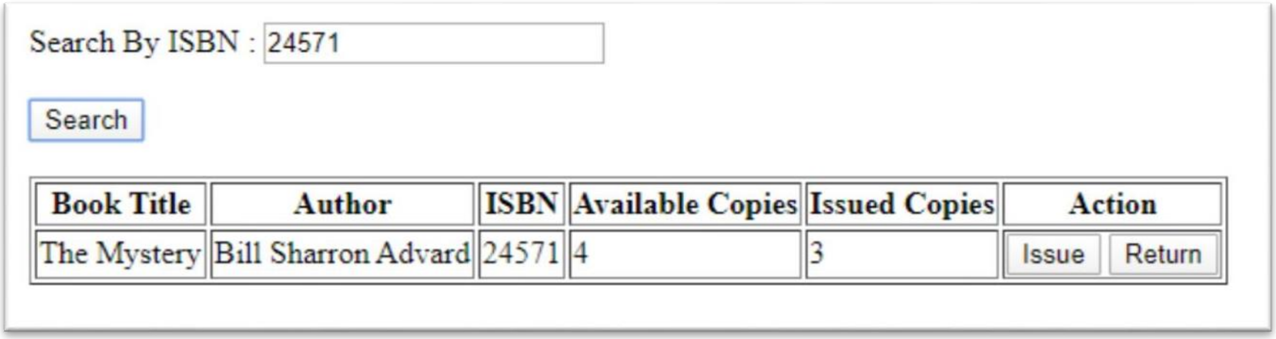
All Books :

Book Title	Author	ISBN	Available Copies	Issued Copies	Action
The Mystery	Bill Sharron Advard	24571	5	2	Issue Return



All Books :

Book Title	Author	ISBN	Available Copies	Issued Copies	Action
The Mystery	Bill Sharron Advard	24571	4	3	Issue Return



Search By ISBN : 24571

Search

Book Title	Author	ISBN	Available Copies	Issued Copies	Action
The Mystery	Bill Sharron Advard	24571	4	3	Issue Return