Omid Mohaghegh Doust- become a back-end expert- Simplilearn project

Front end codes:

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

<%
String LoginName= request.getParameter("fname");
String LoginName1= request.getSession().getAttribute("FName").toString();

if (LoginName!= null){

        %>
        <h1>Welcome to Hibernate  <%= LoginName %>!</h1>
         <% }else{

                %>
                <h1>Welcome to Hibernate  <%= LoginName1 %>!</h1>
        <% }
         %>




        <br/>
        <br/>
        <br/>

         <a href="Master_List.html"> Go to Master List for Set up data</a>

        <br/>
        <br/>
        <br/>
         <a href="ClassSubjectAssign"> Go to Assign Page</a>

        <br/>
        <br/>
        <br/>

                <a href="ShowStudent">Show Student Report </a>
        <br/>
        <br/>
        <br/>


                <a href="ClassShow">Show Class Report </a>
        <br/>
        <br/>
```

```
       <br/>

             <a href="SubjectShow">Show Subject Report </a>
       <br/>
       <br/>
       <br/>

             <a href="TeacherShow">Show Teacher Report </a>
       <br/>
<br>
<br>
<br>
<a href="logout.jsp" >Log Out</a>

</body>
</html>
```

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

<form action="LoginRegister" method="POST">

Enter Your First Name <input type="text" name="fname" maxlength=20>
Enter Your Last Name <input type="text" name="lname" maxlength=40><br>
Enter Your Login Name <input type="text" name="ulogin">
Enter Your Password <input type="text" name="userpassword">
<button>Submit</button>

</form>


</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Master List Page</title>
</head>
<body>

<div align= left>

             <h2>Set up Class Master List</h2>
             <form action="ClassRegister" method="POST">
                    <!-- Enter Class ID: <input type="number" name="Class_ID"/><br/>
-->
                    Enter Class Name: <input type="text" name="Class_Name"/>
                    Enter Class Level: <input type="text" name="Class_Level"/>
                    Enter Class Type: <input type="text" name="Class_Type"/>


                    <input type="submit" value="Set_up">
             </form>

       </div>
<br/>
<br/>

<div align= left>

             <h2>Set up Teacher Master List</h2>
             <form action="TeacherRegister" method="POST">
                    <!-- Enter Teacher ID: <input type="number"
name="Teacher_ID"/><br/>  -->
                    Enter Teacher First Name: <input type="text"
name="Teacher_FName"/>
                    Enter Teacher Last Name: <input type="text"
name="Teacher_LName"/><br/>
                    Enter Teacher Expertise: <input type="text"
name="Teacher_Expert"/>
                    Enter Teacher Level: <input type="text" name="Teacher_Level"/>

                    <input type="submit" value="Set_up">
             </form>

       </div>

       <br/>
<br/>

<div align= left>

             <h2>Set up Subject Master List</h2>
             <form action="SubjectRegister" method="POST">
                    <!-- Enter Subject ID: <input type="number"
name="Subject_ID"/><br/>  -->
                    Enter Subject Name: <input type="text" name="Subject_Name"/>
                    Enter Subject Level: <input type="text" name="Subject_Level"/>
                    Enter Subject Category: <input type="text" name="Subject_Cat"/>
                    <input type="submit" value="Set_up">
             </form>

       </div>

       <br/>
<br/>

<div align= left>
```

```
             <h2>Set up Student Master List</h2>
             <form action="StudentRegister" method="POST">
                    <!-- Enter Student ID: <input type="number"
name="Student_ID"/><br/> -->
                    Enter Student First Name: <input type="text"
name="Student_FName"/>
                    Enter Student Last Name: <input type="text"
name="Student_LName"/> <br/>
                    Enter Student Phone Number: <input type="number"
name="Student_Phone"/>
                    Enter Student Address: <input type="text"
name="Student_Address"/><br/>
                    Enter Student Email: <input type="text" name="Student_Email"/>
                    Enter Student Level: <input type="text" name="Student_Level"/>

                    <input type="submit" value="Set_up">
             </form>

       </div>

       <br/>
<br/>

</body>
</html>



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

<%
session.invalidate();
%>
<br>
<h2>You Logged out successfully!</h2>
<br>
<br>
<br>
<br>
<a href="index.jsp">
Click here to login again!
</a>

</body>
</html>

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

```
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Login Check Page</title>
</head>
<body>

<%
	Object userlogin=request.getSession().getAttribute("UserLog");
	Object userpass= request.getSession().getAttribute("UserPass");
	Object Ulogin= request.getSession().getAttribute("UserLog1");
	Object Passwd= request.getSession().getAttribute("UserPass1");


	if (Ulogin.equals(userlogin) && Ulogin!=null && Passwd.equals(userpass) &&
Passwd!=null ){

		response.sendRedirect("start.jsp");

	} else
	{
		response.sendRedirect("index.jsp?error=1");

	}

%>



</body>
</html>
```

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

<%
if (request.getParameter("error")!= null){

	out.println("Invalid Credential! Please Try Again!");
}

%>
```

```
<form action="LoginRegister" method="GET">

Enter Login <input type="text"  name="ulogin" maxlength=40>
Enter Password <input type="password" name="passwd" maxlength=20>
<button>Log In</button>
</form>

<a href="Sign Up.jsp">Sign Up</a>


</body>
</html>



<%@ 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">
<%@ page import="java.util.List" %>
<%@ page import="java.util.ArrayList" %>
<%@ page import="com.entities.Subject" %>
<%@ page import="com.entities.Schoolcls" %>
<%@ page import="com.entities.Teacher" %>
<%@ page import="com.entities.Student" %>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Assign Page</title>
</head>
<body>

<%
      Object Result=request.getSession().getAttribute("AllSubjects");
      Object Result1=request.getSession().getAttribute("AllClasses");
      Object Result2=request.getSession().getAttribute("AllTeachers");
      Object Result3=request.getSession().getAttribute("AllStudents");


         if (Result!=null || Result1!=null || Result2!=null || Result3!=null ){

             List<Schoolcls> AllClasses= (List<Schoolcls>) Result1;
             List<Subject> AllSubjects=(List<Subject>) Result;
             List<Teacher> AllTeachers= (List<Teacher>) Result2;
             List<Student> AllStudents= (List<Student>) Result3;

                                %>

       <h1>Assign Subjects to Classes</h1>

<form action="ClassSubjectAssign" method="post">
<table align="left">

      <tr>
                   <td><select name="SubjectName">
```

```
                    <%for(Subject  sub1: AllSubjects){
                          if(sub1.getFltr_su()!= null){%>
                    <option value="<%=sub1.getName_su()%>"><%= sub1.getName_su()
%></option>
                    <% }
                    }%>
                    </select>
                    </td>
                    <td>Subjects</td>

       </tr>
       <tr>
                    <td><select name="ClassName">
                    <%for(Schoolcls  c: AllClasses){
                          if(c.getFltr()!= null){%>
                    <option value="<%=c.getName_cl() %>"><%= c.getName_cl()
%></option>
                    <% }

                          }%>
                    </select>
                    </td>
                    <td>Classes</td>

       </tr>


       <tr>
              <td align="center" colspan="2"><input type="submit"
value="Assign!"></td>
        </tr>
</table>

</form>

<br/>
<br/>
<br/>

 <h1>Assign Teachers to Classes to Subjects </h1>

<form action="TeacherAssignControl" method="post">
<table align="left">

       <tr>
                    <td><select name="TeacherName">
                    <%for(Teacher  t: AllTeachers){
                          if(t.getFltrt()!= null){%>
                    <option value="<%=t.getLname_tc()%>"><%= t.getLname_tc()
%></option>
                    <% }
                    }%>
                    </select>
                    </td>
                    <td>Teachers</td>
```

```
      </tr>
      <tr>
                   <td><select name="ClassName">
                   <%for(Schoolcls  cl1: AllClasses){
                         if(cl1.getFltr()!= null){%>
                   <option value="<%=cl1.getName_cl() %>"><%= cl1.getName_cl()
%></option>
                   <% }

                         }%>
                   </select>
                   </td>
                   <td>Classes</td>

      </tr>

      <tr>
                   <td><select name="SubjectName">
                   <%for(Subject  sub2: AllSubjects){
                         if(sub2.getFltr_su()!= null){%>
                   <option value="<%=sub2.getName_su()%>"><%= sub2.getName_su()
%></option>
                   <% }
                   }%>
                   </select>
                   </td>
                   <td>Subjects</td>

      </tr>


      <tr>
              <td align="center" colspan="2"><input type="submit"
value="Assign!"></td>
       </tr>
</table>

</form>

<br/>
<br/>
<br/>
<br/>
<br/>
 <h1>Assign Students to Classes</h1>
<form action="StudentAssignControl" method="post">
<table align="left">

      <tr>
                   <td><select name="StudentName">
                   <%for(Student  std: AllStudents){
                         if(std.getFltrs()!= null){%>
                   <option value="<%=std.getLname_std()%>"><%= std.getLname_std()
%></option>
```

```
                    <% }
                    }%>
                    </select>
                    </td>
                    <td>Students</td>

       </tr>
       <tr>
                    <td><select name="ClassName">
                    <%for(Schoolcls  cl2: AllClasses){
                          if(cl2.getFltr()!= null){%>
                    <option value="<%=cl2.getName_cl() %>"><%= cl2.getName_cl()
%></option>
                    <% }

                          }%>
                    </select>
                    </td>
                    <td>Classes</td>

       </tr>


       <tr>
              <td align="center" colspan="2"><input type="submit"
value="Assign!"></td>
        </tr>
</table>

</form>


<%
          }
%>


</body>
</html>




======================================================================================
======================================================================================
```

}

Back-end codes:

```
package com.connection;


import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;


public class DBConnection {


        private Connection connection;


        public DBConnection(String dbURL, String user, String pwd) throws ClassNotFoundException,
SQLException {

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

                this.connection = DriverManager.getConnection(dbURL, user, pwd);


        }


        public Connection getConnection() {

                return this.connection;

        }


        public void closeConnection() throws SQLException {

                if (this.connection != null)

                        this.connection.close();

        }


}
```

```java
package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;

import com.entities.Teacher;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class TeacherShowController
 */
@WebServlet("/TeacherShow")
public class TeacherShowController extends HttpServlet {
	private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public TeacherShowController() {
      super();

   }

        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        @SuppressWarnings("unchecked")
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

                PrintWriter out = response.getWriter();
                out.println("<html>" + "<head>" + "<style>" + "table, th, td {" + "  border: 1px solid black;" + "}"
                        + "</style>" + "</head>" + "<body>");
                out.println("<h1>Teacher Report</h1>");

                SessionFactory factory = hibernateUtil.getSessionFactory();

                Session session = factory.openSession();

                try {

                        List<Teacher> list = session.createQuery("from Teacher").list();

                        if (list != null && list.size() > 0) {


                                out.println("<table style=\"width:100%\">");
```

```
                    out.println("<tr>" +  "    <th>Teacher-Fistname</th>" + "    <th>Teacher-
Lastname</th>"

                              + "    <th>Teacher-Expertise</th>" + "    <th>Teacher-
Level</th>"+

                               "  </tr>");

                    out.println("<h2> This is a list of available subjects </h2>");



                    for (Teacher tc1 : list) {
                         if(tc1.getFltrt() != null) {
                         out.println("<tr>");

                         out.println("<td>" + tc1.getFname_tc() + "</td>");
                         out.println("<td>" + tc1.getLname_tc() + "</td>");
                         out.println("<td>" + tc1.getExpertise_tc() + "</td>");
                         out.println("<td>" + tc1.getLevel_tc() + "</td>");



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

                    out.println("<table style=\"width:100%\">");
                    out.println("<tr>" +  "    <th>Teacher-Fistname</th>" + "    <th>Teacher-
Lastname</th>"

                              + "    <th>Teacher-Expertise</th>" + "    <th>Teacher-
Level</th>"+

                              "    <th>Teacher-Class</th>"+ "  </tr>");
```

```
            out.println("<h2> This is a list of registered teachers to classes </h2>");



            for (Teacher tc2 : list) {
                if(tc2.getFltrt() == null) {
                out.println("<tr>");


                out.println("<td>" + tc2.getFname_tc() + "</td>");
                out.println("<td>" + tc2.getLname_tc() + "</td>");
                out.println("<td>" + tc2.getExpertise_tc() + "</td>");
                out.println("<td>" + tc2.getLevel_tc() + "</td>");
                out.println("<td>" + tc2.getClasses() + "</td>");


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




        out.println("</body></html>");


    }catch (Exception ex) {
        throw ex;
    } finally {
        session.close();
    }


}
```

```
	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

		doGet(request, response);
	}

}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;
//import java.util.HashSet;
//import java.util.Set;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import com.entities.Teacher;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class TeacherRegisterController
 */
@WebServlet("/TeacherRegister")
public class TeacherRegisterController extends HttpServlet {
	private static final long serialVersionUID = 1L;

   /**
    * @see HttpServlet#HttpServlet()
    */
   public TeacherRegisterController() {
      super();

   }

	/**
	 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

		PrintWriter out = response.getWriter();
		out.println("<html><body>");
		System.out.println("servlet started!");

		SessionFactory factory = hibernateUtil.getSessionFactory();
```

```
        Session session = factory.openSession();
        out.println("Hibernate Session opened.<br>");

            Transaction transaction = null;
        try {
            transaction = session.beginTransaction();
            out.println("Transaction began.<br>");

            String teacherFname= request.getParameter("Teacher_FName");
            String teacherLname= request.getParameter("Teacher_LName");
            String teacherExpert= request.getParameter("Teacher_Expert");
            String teacherLevel= request.getParameter("Teacher_Level");

            Teacher teacher= new Teacher( teacherFname, teacherLname, teacherExpert,
teacherLevel, "Ok");

            session.save(teacher);
            transaction.commit();
            out.println("Record with name : "+teacherFname +" Saved successfully." );

            out.println("Hibernate Session closed.<br>");
            out.println("</body></html>");
        } catch (Exception ex) {
            ex.printStackTrace();
            System.err.println("There is an error in hinernate functionality!");

        } finally {
                session.close();
            }
        }

        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

            doPost(request, response);
        }

}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.HashSet;
import java.util.List;
import java.util.Set;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import com.entities.Schoolcls;
import com.entities.Subject;
import com.entities.Teacher;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class TeacherAssignController
 */
@WebServlet("/TeacherAssignControl")
public class TeacherAssignController extends HttpServlet {
	private static final long serialVersionUID = 1L;

   /**
    * @see HttpServlet#HttpServlet()
    */
   public TeacherAssignController() {
      super();

   }

	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
	 */
	@SuppressWarnings("unchecked")
```

```
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {


        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        System.out.println("servlet started!");

        SessionFactory factory = hibernateUtil.getSessionFactory();

        Session session = factory.openSession();
        out.println("Hibernate Session opened.<br>");

            Transaction transaction = null;
        try {
            transaction = session.beginTransaction();
            out.println("Transaction began.<br>");

            Object allClasses= request.getSession().getAttribute("AllClasses");
            Object allSubjects = request.getSession().getAttribute("AllSubjects");
            Object allTeachers= request.getSession().getAttribute("AllTeachers");



            List<Schoolcls> AllClss= (List<Schoolcls>) allClasses;
            List<Subject> AllSubjcts=(List<Subject>) allSubjects;
            List<Teacher> AllTeachs=(List<Teacher>) allTeachers;





            String className= request.getParameter("ClassName");
```

```
Schoolcls cls1 = new Schoolcls();

for (Schoolcls c1: AllClss) {
if(c1.getName_cl()!=null && c1.getName_cl().equals(className)) {
    String clsnm= c1.getName_cl();
    String clslev= c1.getLevel_cl();
    String clstp= c1.getType_cl();
    cls1.setName_cl(clsnm);
    cls1.setLevel_cl(clslev);
    cls1.setType_cl(clstp);
    cls1.setFltr(null);
    }
}


String subjectName= request.getParameter("SubjectName");
Set<Subject> subjects = new HashSet<Subject>();
Subject sub= new Subject();

for (Subject s1: AllSubjcts) {
if(s1.getName_su()!=null && s1.getName_su().equals(subjectName)) {
String subnm = s1.getName_su();
String sublev= s1.getLevel_su();
String subcat= s1.getCat_su();
sub.setName_su(subnm);
sub.setLevel_su(sublev);
sub.setCat_su(subcat);
        }
}
```

```
subjects.add(sub);



String teacherName= request.getParameter("TeacherName");
Set<Teacher> teachers = new HashSet<Teacher>();
Teacher teach= new Teacher();

for (Teacher t1: AllTeachs) {
if(t1.getLname_tc()!=null && t1.getLname_tc().equals(teacherName)) {
String tfnm = t1.getFname_tc();
String tlev= t1.getLevel_tc();
String texp= t1.getExpertise_tc();
teach.setFname_tc(tfnm);
teach.setLname_tc(teacherName);
teach.setLevel_tc(tlev);
teach.setExpertise_tc(texp);
                }
}

teachers.add(teach);



out.println("Class recognized successfully!");

cls1.setSubjects(subjects);
cls1.setTeachers(teachers);
```

```
            }

            session.saveOrUpdate(cls1);

            transaction.commit();

            out.println("Record with Subject ID : "+subjectName +" Saved successfully." );


            out.println("Hibernate Session closed.<br>");

            out.println("</body></html>");


        } catch (Exception ex) {


            ex.printStackTrace();

            System.err.println("There is an error in hinernate functionality!");


        } finally {


            session.close();

        }


    }




    /**

     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

     */

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

        doPost(request, response);


            }

        }
```

```
package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;

import com.entities.Subject;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class SubjectShowController
 */
@WebServlet("/SubjectShow")
public class SubjectShowController extends HttpServlet {
	private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public SubjectShowController() {
        super();

    }

            /**
             * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
             */
            @SuppressWarnings("unchecked")
            protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

                        PrintWriter out = response.getWriter();
                        out.println("<html>" + "<head>" + "<style>" + "table, th, td {" + "  border: 1px solid
black;" + "}"
                                    + "</style>" + "</head>" + "<body>");
                        out.println("<h1>Subject Report</h1>");
                        SessionFactory factory = hibernateUtil.getSessionFactory();

                        Session session = factory.openSession();

                        try {

                                    List<Subject> list = session.createQuery("from Subject").list();

                                    if (list != null && list.size() > 0) {

                                                out.println("<table style=\"width:100%\">");
                                                out.println("<tr>" +  "    <th>Subject-Name</th>" + "    <th>Subject-
Level</th>"
                                                            + "    <th>Subject-Category</th>" +  "  </tr>");
```

```
                out.println("<h2> This is a list of available subjects </h2>");


                for (Subject sj : list) {
                    if(sj.getFltr_su() != null) {
                    out.println("<tr>");
                    out.println("<td>" + sj.getName_su() + "</td>");
                    out.println("<td>" + sj.getLevel_su() + "</td>");
                    out.println("<td>" + sj.getCat_su() + "</td>");


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




                out.println("<table style=\"width:100%\">");
                out.println("<tr>" +  "    <th>Subject-Name</th>" + "    <th>Subject-
Level</th>"
                        + "    <th>Subject-Category</th>" + "
<th>Subject_Class</th>" +  "  </tr>");


                out.println("<h2> This is a list of registered subjects to classes </h2>");


                for (Subject sj1 : list) {
                    if(sj1.getFltr_su() == null) {
                    out.println("<tr>");
                    out.println("<td>" + sj1.getName_su() + "</td>");
                    out.println("<td>" + sj1.getLevel_su() + "</td>");
                    out.println("<td>" + sj1.getCat_su() + "</td>");
}


                                        out.println("<td>" + sj1.getClasses() + "</td>");


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

                        }




                        out.println("</body></html>");


                } catch (Exception ex) {
                        throw ex;
                } finally {
                        session.close();
                }


        }


        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {


                doGet(request, response);
        }
}
```

```
package com.control;

import java.io.IOException;
import java.io.PrintWriter;
//import java.util.HashSet;
//import java.util.Set;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import com.entities.Subject;
//import com.entities.Teacher;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class SubjectRegisterController
 */
@WebServlet("/SubjectRegister")
public class SubjectRegisterController extends HttpServlet {
	private static final long serialVersionUID = 1L;

   /**
```

```
    * @see HttpServlet#HttpServlet()
    */
   public SubjectRegisterController() {
      super();

   }

        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

                PrintWriter out = response.getWriter();
                out.println("<html><body>");
                System.out.println("servlet started!");

                SessionFactory factory = hibernateUtil.getSessionFactory();

                Session session = factory.openSession();
                out.println("Hibernate Session opened.<br>");

                        Transaction transaction = null;
                try {
                        transaction = session.beginTransaction();
                        out.println("Transaction began.<br>");

                        String subjectName= request.getParameter("Subject_Name");
                        String subjectCat= request.getParameter("Subject_Cat");
```

```
            String subjectLevel= request.getParameter("Subject_Level");
            String fltrsu= "Ok";

            Subject sub= new Subject( subjectName, subjectLevel, subjectCat, fltrsu);



            session.save(sub);
            transaction.commit();
            out.println("Record with name : "+subjectName +" Saved successfully." );

            out.println("Hibernate Session closed.<br>");
            out.println("</body></html>");
        } catch (Exception ex) {
            ex.printStackTrace();
            System.err.println("There is an error in hinernate functionality!");

        } finally {

            session.close();
        }
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

        doPost(request, response);
        }

}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.HashSet;
import java.util.List;
import java.util.Set;



import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import org.hibernate.query.Query;

import com.entities.Schoolcls;
import com.entities.Subject;
import com.hibernate.hibernateUtil;
```

```
/**
 * Servlet implementation class ClassAssignController
 */
@WebServlet("/ClassSubjectAssign")
public class SubjectAssignController extends HttpServlet {
	private static final long serialVersionUID = 1L;


   /**
    * @see HttpServlet#HttpServlet()
    */
   public SubjectAssignController() {
      super();

   }

	/**
	 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
	 */

	@SuppressWarnings("unchecked")
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

		PrintWriter out = response.getWriter();
		out.println("<html><body>");
		System.out.println("servlet started!");

		SessionFactory factory = hibernateUtil.getSessionFactory();
```

```
Session session = factory.openSession();
out.println("Hibernate Session opened.<br>");

    Transaction transaction = null;
try {
    transaction = session.beginTransaction();
    out.println("Transaction began.<br>");

    Object allClasses= request.getSession().getAttribute("AllClasses");
    Object allSubjects = request.getSession().getAttribute("AllSubjects");


    List<Schoolcls> AllClss= (List<Schoolcls>) allClasses;
    List<Subject> AllSubjcts=(List<Subject>) allSubjects;



    String className= request.getParameter("ClassName");


    Schoolcls cls1 = new Schoolcls();

    for (Schoolcls c1: AllClss) {
    if(c1.getName_cl()!=null && c1.getName_cl().equals(className)) {
        String clsnm= c1.getName_cl();
        String clslev= c1.getLevel_cl();
        String clstp= c1.getType_cl();
        cls1.setName_cl(clsnm);
        cls1.setLevel_cl(clslev);

        cls1.setType_cl(clstp);

        cls1.setFltr(null);


        }

    }



    String subjectName= request.getParameter("SubjectName");

    Set<Subject> subjects = new HashSet<Subject>();

    Subject sub= new Subject();


    for (Subject s1: AllSubjcts) {

    if(s1.getName_su()!=null && s1.getName_su().equals(subjectName)) {

    String subnm = s1.getName_su();

    String sublev= s1.getLevel_su();

    String subcat= s1.getCat_su();

    sub.setName_su(subnm);

    sub.setLevel_su(sublev);

    sub.setCat_su(subcat);

                    }

    }


    subjects.add(sub);





    out.println("Class recognized successfully!");
```

```
            cls1.setSubjects(subjects);


            session.saveOrUpdate(cls1);

            transaction.commit();

            out.println("Record with Subject ID : "+subjectName +" Saved successfully." );


            out.println("Hibernate Session closed.<br>");

            out.println("</body></html>");


        } catch (Exception ex) {


            ex.printStackTrace();

            System.err.println("There is an error in hinernate functionality!");


        } finally {


            session.close();

        }


    }




    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {



        PrintWriter out = response.getWriter();

        out.println("<html><body>");
        System.out.println("doGet servlet started!");
    SessionFactory factory = hibernateUtil.getSessionFactory();


        Session session = factory.openSession();
        out.println("Hibernate Session opened.<br>");


            Transaction transaction = null;
        try {
            transaction = session.beginTransaction();
            out.println("Transaction began.<br>");




                @SuppressWarnings("unchecked")
                Query<Subject> query = session.createQuery("from Subject"
).setCacheable(true);//.setHint("org.hibernate.cacheable","true");
                List<Subject> Result = (List<Subject>) query.getResultList();


                @SuppressWarnings("unchecked")
                Query<Schoolcls> query1 = session.createQuery("from Schoolcls"
).setCacheable(true);//.setHint("org.hibernate.cacheable","true");
                List<Schoolcls> Result1 = (List<Schoolcls>) query1.getResultList();


                @SuppressWarnings("unchecked")
                Query<Subject> query2 = session.createQuery("from Teacher"
).setCacheable(true);//.setHint("org.hibernate.cacheable","true");
                List<Subject> Result2 = (List<Subject>) query2.getResultList();


                @SuppressWarnings("unchecked")
                Query<Subject> query3 = session.createQuery("from Student"
).setCacheable(true);//.setHint("org.hibernate.cacheable","true");
```

```
            List<Subject> Result3 = (List<Subject>) query3.getResultList();


            request.getSession().setAttribute("AllSubjects", Result);
            request.getSession().setAttribute("AllClasses", Result1);
            request.getSession().setAttribute("AllTeachers", Result2);
            request.getSession().setAttribute("AllStudents", Result3);
            response.sendRedirect("Assign.jsp");




            transaction.commit();
            out.println("Record with Result and Result1 received successfully." );


            out.println("Hibernate doGet Session closed.<br>");
            out.println("</body></html>");
        } catch (Exception ex) {


            ex.printStackTrace();
            System.err.println("There is an error in hinernate functionality!");


        } finally {


            session.close();
        }


}


/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
```

```
          */



}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;

import com.entities.Student;
import com.hibernate.hibernateUtil;



/**
 * Servlet implementation class StudentShowController
 */
@WebServlet("/ShowStudent")
public class StudentShowController extends HttpServlet {
```

```
          private static final long serialVersionUID = 1L;

   /**
    * @see HttpServlet#HttpServlet()
    */
   public StudentShowController() {
      super();

   }

          /**
           * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
           */

          @SuppressWarnings("unchecked")
          protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

                    PrintWriter out = response.getWriter();
                    out.println("<html>" + "<head>" + "<style>" + "table, th, td {" + "  border: 1px solid
black;" + "}"
                                        + "</style>" + "</head>" + "<body>");
                    out.println("<h1>Student Report</h1>");
                    SessionFactory factory = hibernateUtil.getSessionFactory();

                    Session session = factory.openSession();
```

```
        try {

            List<Student> list = session.createQuery("from Student").list();

            if (list != null && list.size() > 0) {

                out.println("<table style=\"width:100%\">");
                out.println("<tr>" + "    <th>Student_FirstName</th>" + "
<th>Student_LastName</th>"
                        + "    <th>Student_PhoneNumber</th>"+ "
<th>Student_Address</th>"+ "    <th>Student_Email</th>" +
                        "  </tr>");
                out.println("<h2> This is a list of available students </h2>");


                for (Student st5 : list) {

                    if(st5.getFltrs() != null) {
                    out.println("<tr>");

                    out.println("<td>" + st5.getFname_std() + "</td>");
                    out.println("<td>" + st5.getLname_std() + "</td>");
                    out.println("<td>" + st5.getNumber_std() + "</td>");
                    out.println("<td>" + st5.getSaddress() + "</td>");
                    out.println("<td>" + st5.getSemail() + "</td>");

                    out.println("</tr>");
                    }
                }
                out.println("<table style=\"width:100%\">");
                out.println("<tr>" + "    <th>Student_FirstName</th>" + "
<th>Student_LastName</th>"
                        + "    <th>Student_PhoneNumber</th>"+ "
<th>Student_Address</th>"+ "    <th>Student_Email</th>" +
                        "    <th>Student_Class</th>" + "  </tr>");
                out.println("<h2> This is a list of registered students to classes </h2>");


                for (Student st6 : list) {

                    if(st6.getFltrs() == null) {
                    out.println("<tr>");

                    out.println("<td>" + st6.getFname_std() + "</td>");
                    out.println("<td>" + st6.getLname_std() + "</td>");
                    out.println("<td>" + st6.getNumber_std() + "</td>");
                    out.println("<td>" + st6.getSaddress() + "</td>");
                    out.println("<td>" + st6.getSemail() + "</td>");

                    out.println("<td>" + st6.getSch_class().getName_cl() + "</td>");


                    out.println("</tr>");
                            }
                        }
                    }
```

```
            out.println("</body></html>");
        } catch (Exception ex) {
            throw ex;
        } finally {
            session.close();
        }
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        doGet(request, response);
    }

}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import com.entities.Schoolcls;
import com.entities.Student;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class StudentController
 */
@WebServlet("/StudentRegister")
public class StudentRegisterController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	/**
	 * Default constructor.
	 */
	public StudentRegisterController() {

	}

	/**
	 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
	 *      response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response)
			throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        System.out.println("servlet started!");

        SessionFactory factory = hibernateUtil.getSessionFactory();

        Session session = factory.openSession();
        out.println("Hibernate Session opened.<br>");

        Transaction transaction = null;
        try {
            transaction = session.beginTransaction();
            out.println("Transaction began.<br>");

            String studentFname = request.getParameter("Student_FName");
            String studentLname = request.getParameter("Student_LName");
            String studentPhone = request.getParameter("Student_Phone");
            String studentAddress = request.getParameter("Student_Address");
            String studentEmail = request.getParameter("Student_Email");
            String studentLevel = request.getParameter("Student_Level");
            String fltrs = "Ok";

            Schoolcls clss = new Schoolcls();
            clss.setStudents(null);

            Student student = new Student(studentFname, studentLname,
Integer.parseInt(studentPhone), studentAddress,
                    studentEmail, studentLevel, fltrs, clss);
```

```
			session.save(student);
			transaction.commit();
			out.println("Record with name : " + studentFname + " Saved successfully.");

			out.println("Hibernate Session closed.<br>");
			out.println("</body></html>");
		} catch (Exception ex) {
			ex.printStackTrace();
			System.err.println("There is an error in hinernate functionality!");

		} finally {

			session.close();
		}
	}

	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
	 *      response)
	 */
	protected void doGet(HttpServletRequest request, HttpServletResponse response)
			throws ServletException, IOException {

		doPost(request, response);
	}

}
```

```
package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.HashSet;
import java.util.List;
import java.util.Set;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;


import com.entities.Schoolcls;
import com.entities.Student;

import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class StudentAssignController
 */
@WebServlet("/StudentAssignControl")
public class StudentAssignController extends HttpServlet {
        private static final long serialVersionUID = 1L;


   /**
    * @see HttpServlet#HttpServlet()
    */
   public StudentAssignController() {
       super();


   }


        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
         */
        @SuppressWarnings("unchecked")
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
                PrintWriter out = response.getWriter();
                out.println("<html><body>");
                System.out.println("servlet started!");


                SessionFactory factory = hibernateUtil.getSessionFactory();


                Session session = factory.openSession();
                out.println("Hibernate Session opened.<br>");


                        Transaction transaction = null;
                try {
                        transaction = session.beginTransaction();
                        out.println("Transaction began.<br>");
```

```
Object allClasses= request.getSession().getAttribute("AllClasses");

Object allStudents= request.getSession().getAttribute("AllStudents");

List<Schoolcls> AllClss= (List<Schoolcls>) allClasses;

List<Student> AllStuds=(List<Student>) allStudents;


String studentName= request.getParameter("StudentName");

Set<Student> students = new HashSet<Student>();
Student stud= new Student();

for (Student s2: AllStuds) {
if(s2.getLname_std()!=null && s2.getLname_std().equals(studentName)) {

String sfnm = s2.getFname_std();
String slev= s2.getSlevel();
long sphn= s2.getNumber_std();
String stemail= s2.getSemail();
String stadd= s2.getSaddress();
stud.setFname_std(sfnm);
stud.setNumber_std(sphn);
stud.setSaddress(stadd);
stud.setSlevel(slev);
stud.setLname_std(studentName);
stud.setSemail(stemail);
```

```
        }
}

students.add(stud);

String className= request.getParameter("ClassName");

Schoolcls cls1 = new Schoolcls();

for (Schoolcls c1: AllClss) {
if(c1.getName_cl()!=null && c1.getName_cl().equals(className)) {

    String clsnm= c1.getName_cl();
    String clslev= c1.getLevel_cl();
    String clstp= c1.getType_cl();
    cls1.setName_cl(clsnm);
    cls1.setLevel_cl(clslev);
    cls1.setType_cl(clstp);
    cls1.setFltr(null);
    }
}



out.println("Class recognized successfully!");


cls1.setStudents(students);
```

```
}


                    stud.setSch_class(cls1);

                    session.saveOrUpdate(stud);

                    session.saveOrUpdate(cls1);

                    transaction.commit();

                    out.println("Record with Subject ID : "+studentName +" Saved successfully." );


                    out.println("Hibernate Session closed.<br>");

                    out.println("</body></html>");


            } catch (Exception ex) {


                    ex.printStackTrace();

                    System.err.println("There is an error in hinernate functionality!");


            } finally {


                    session.close();

            }


        }


        /**

         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

         */

        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

                doPost(request, response);

                }

}
```

```
package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;


import com.hibernate.hibernateUtil;
import com.loginentity.Login;

/**
 * Servlet implementation class LoginController
 */
@WebServlet("/LoginRegister")
public class LoginRegisterController extends HttpServlet {
        private static final long serialVersionUID = 1L;

   /**
    * @see HttpServlet#HttpServlet()
    */
   public LoginRegisterController() {
       super();


   }


   /**
             * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
             */
            protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {



                        PrintWriter out = response.getWriter();
                        out.println("<html><body>");
                        System.out.println("servlet started!");

                        SessionFactory factory = hibernateUtil.getSessionFactory();

                        Session session = factory.openSession();

                        Transaction transaction = null;
                        try {
                                    transaction = session.beginTransaction();

                                    String Fname = request.getParameter("fname");
                                    String Lname = request.getParameter("lname");
                                    String login = request.getParameter("ulogin");
                                    String password = request.getParameter("userpassword");
                                    Login userLogin = new Login(Fname, Lname, login, password);

                                    session.saveOrUpdate(userLogin);
                                    transaction.commit();

                                    out.println("</body></html>");

                                    request.getSession().setAttribute("FName", Fname);



                        } catch (Exception ex) {
                                    ex.printStackTrace();
                                    System.err.println("There is an error in hinernate functionality!");

                        } finally {

                                    session.close();

                                    response.sendRedirect("start.jsp");
                        }
            }
            /**
             * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
             */
            @SuppressWarnings("unchecked")
            protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
```

```
PrintWriter out= response.getWriter();
out.println("<html><body>");


SessionFactory factory= hibernateUtil.getSessionFactory();
Session session= factory.openSession();


Transaction transaction=null;
 try {
        transaction = session.beginTransaction();


        List<Login> logininfo= session.createQuery("from Login").list();


        if(logininfo!=null && logininfo.size()>0) {



        String Userlog1= request.getParameter("ulogin");
        String Userpass1= request.getParameter("passwd");
        request.getSession().setAttribute("UserLog1", Userlog1);
        request.getSession().setAttribute("UserPass1", Userpass1);



        for (Login lg:logininfo) {
                 if (lg.getUlogin().equals(Userlog1)) {
                String userLog=lg.getUlogin();
                String userPass= lg.getUpassword();
                String firstnm= lg.getUfname();


                request.getSession().setAttribute("UserLog", userLog);
                request.getSession().setAttribute("UserPass", userPass);
                request.getSession().setAttribute("FName", firstnm);
                 }
                }
            }


            transaction.commit();


        }
        catch(Exception ex) {

            ex.printStackTrace();
            System.err.println("Error in hibernate functionality!");

        }finally {

            ex.close();
```

```
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;

import com.entities.Schoolcls;
import com.hibernate.hibernateUtil;

/**
 * Servlet implementation class ClassShowController
 */
@WebServlet("/ClassShow")
public class ClassShowController extends HttpServlet {
	private static final long serialVersionUID = 1L;


    /**
     * @see HttpServlet#HttpServlet()
     */
    public ClassShowController() {
        super();
```

```
   }


           /**
            * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
            */
           @SuppressWarnings("unchecked")
           protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {



                      PrintWriter out = response.getWriter();
                      out.println("<html>" + "<head>" + "<style>" + "table, th, td {" + "  border: 1px solid
black;" + "}"
                                          + "</style>" + "</head>" + "<body>");
                      out.println("<h1>Class Report</h1>");
                      SessionFactory factory = hibernateUtil.getSessionFactory();


                      Session session = factory.openSession();


                      try {


                                 List<Schoolcls> list = session.createQuery("from Schoolcls").list();


                                 if (list != null && list.size() > 0) {



                                                                out.println("<table style=\"width:100%\">");
                                                                out.println("<tr>" +  "    <th>Class-name</th>" + "
<th>Class-level</th>"
                                                                           + "    <th>Class-type</th>" + "  </tr>");
```

```
                        out.println("<h2> This is a list of available classes
</h2>");

                    for (Schoolcls cls: list) {

                        if(cls.getFltr() != null) {



                            out.println("<tr>");


                            out.println("<td>" + cls.getName_cl() + "</td>");
                            out.println("<td>" + cls.getLevel_cl() + "</td>");
                            out.println("<td>" + cls.getType_cl() + "</td>");
                            out.println("</tr>");}
                    }




                            out.println("<table style=\"width:100%\">");
                            out.println("<tr>" +  "    <th>Class-name</th>" +
"    <th>Class-level</th>"
                                    + "    <th>Class-type</th>" + "
<th>Class-subject</th>" + "    <th>Class-teacher</th>" +
                                    "    <th>Class-student</th>" + "
</tr>");
                            out.println("<h2> This is a list of registered
subjects, teachers, and students to the classes </h2>");

                                for (Schoolcls cls1:list) {
                                    if(cls1.getFltr() == null &&
cls1.getName_cl()!= null) {


                                out.println("<tr>");


                                out.println("<td>" + cls1.getName_cl() +
"</td>");
                                out.println("<td>" + cls1.getLevel_cl() +
"</td>");
                                out.println("<td>" + cls1.getType_cl() +
"</td>");
                                out.println("<td>" + cls1.getSubjects() +
"</td>");
                                out.println("<td>" + cls1.getTeachers() +
"</td>");
                                out.println("<td>" + cls1.getStudents() +
"</td>");
                                out.println("</tr>");
                        }
                    }
                }
            out.println("</body></html>");
        }
         catch (Exception ex) {
            throw ex;
        } finally {
            session.close();
        }


    }
```

```
	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

		doGet(request, response);
	}

}

package com.control;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.HashSet;
import java.util.Set;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import com.entities.Schoolcls;

import com.entities.Student;

import com.hibernate.hibernateUtil;


/**

 * Servlet implementation class ClassRegisterController1

 */

@WebServlet("/ClassRegister")

public class ClassRegisterController extends HttpServlet {

    private static final long serialVersionUID = 1L;


   /**

    * @see HttpServlet#HttpServlet()

    */

   public ClassRegisterController() {

      super();


   }


    /**

     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

     */

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {


        PrintWriter out = response.getWriter();

        out.println("<html><body>");

        System.out.println("servlet started!");
```

```
SessionFactory factory = hibernateUtil.getSessionFactory();


Session session = factory.openSession();
out.println("Hibernate Session opened.<br>");


        Transaction transaction = null;
try {
        transaction = session.beginTransaction();
        out.println("Transaction began.<br>");



        Set<Student> students1 = new HashSet<Student>();
        Student stdnt= new Student();


        students1.add(stdnt);


        String className= request.getParameter("Class_Name");
        String classLevel= request.getParameter("Class_Level");
        String classType= request.getParameter("Class_Type");
        String fltr= "Ok";



        Schoolcls cls = new Schoolcls( className , classLevel, classType , fltr);


        session.saveOrUpdate(cls);


        transaction.commit();
        out.println("Record with name : "+className +" Saved successfully." );
```

```
			out.println("Hibernate Session closed.<br>");

			out.println("</body></html>");

		} catch (Exception ex) {


			ex.printStackTrace();

			System.err.println("There is an error in hinernate functionality!");


		} finally {


			session.close();

		}

	}


	/**

	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

	 */

	protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {


		doPost(request, response);

	}


}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
"-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
  <session-factory>
    <!-- Database connection settings -->
    <property name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property
name="connection.url">jdbc:mysql://localhost:3306/hibernatedemo</property>
    <property name="connection.username">root</property>
    <property name="connection.password">Electrical665</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.format_sql">true</property>
    <property name="hibernate.hbm2ddl.auto">update</property>

   </session-factory>
</hibernate-configuration>
```

```
package com.loginentity;


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="Login_Table")
public class Login {


	@Id
	@GeneratedValue(strategy=GenerationType.AUTO)
	@Column(name="Login_Id")
	private int lgid;
	@Column(name="User_FirstName")
	private String ufname;
	@Column(name="User_LastName")
	private String ulname;
	@Column(name="User_Login")
```

```
private String ulogin;
@Column(name="User_Password")
private String upassword;



public Login() {

}



public Login(String ufname, String ulname, String ulogin, String upassword) {

        this.ufname = ufname;
        this.ulname = ulname;
        this.ulogin = ulogin;
        this.upassword = upassword;
}



public int getLgid() {
        return lgid;
}



public void setLgid(int lgid) {
```

```
            this.lgid = lgid;

}




public String getUfname() {

            return ufname;

}




public void setUfname(String ufname) {

            this.ufname = ufname;

}




public String getUlname() {

            return ulname;

}




public void setUlname(String ulname) {

            this.ulname = ulname;

}
```

```
    public String getUlogin() {

        return ulogin;

    }




    public void setUlogin(String ulogin) {

        this.ulogin = ulogin;

    }




    public String getUpassword() {

        return upassword;

    }




    public void setUpassword(String upassword) {

        this.upassword = upassword;

    }




    @Override

    public String toString() {

        return "Login [lgid=" + lgid + ", ufname=" + ufname + ", ulname=" + ulname + ", ulogin="
+ ulogin

                + ", upassword=" + upassword + "]";
    }
}

package com.hibernate;

import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

import com.entities.Schoolcls;
import com.entities.Student;
import com.entities.Subject;
import com.entities.Teacher;
import com.loginentity.Login;


public class hibernateUtil {

    private static final SessionFactory sessionFactory = buildSessionFactory();

    private static SessionFactory buildSessionFactory() {

        try {

            SessionFactory sessionFactory = new
Configuration().configure("hibernate.cfg.xml")

                    .addAnnotatedClass(Student.class)
                    .addAnnotatedClass(Schoolcls.class)
                    .addAnnotatedClass(Subject.class)
```

```
                .addAnnotatedClass(Teacher.class)
                .addAnnotatedClass(Login.class)
                    .buildSessionFactory();


            return sessionFactory;

        } catch (Throwable ex) {

            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }

}

package com.entities;

import java.util.List;

import javax.persistence.Column;
//import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
```

```
//import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.ManyToMany;
import javax.persistence.Table;

@Entity
@Table( name= "Teacher_Table")
public class Teacher {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    @Column(name = "Teacher_Id")
    private long tid;
    @Column(name = "First_Name")
    private String fname_tc;
    @Column(name = "Last_Name")
    private String lname_tc;
    @Column(name = "Teacher_Expertise")
    private String expertise_tc; //Teacher Subject Matter expertise
    @Column(name = "Teacher_Level")
    private String level_tc; //Teacher level for subject level or Class Level
    @Column(name="Filter_T")
    private String fltrt;



    @ManyToMany(mappedBy= "teachers")
    private List<Schoolcls> classes;

    public Teacher() {
        }



    public Teacher(String fname_tc, String lname_tc, String expertise_tc, String level_tc, String fltrt)
{

        this.fname_tc = fname_tc;
        this.lname_tc = lname_tc;
        this.expertise_tc = expertise_tc;
        this.level_tc = level_tc;
        this.fltrt = fltrt;
    }



    public long getTid() {
        return tid;
    }


    public void setTid(long tid) {
        this.tid = tid;
    }


    public String getFname_tc() {
        return fname_tc;

}



public void setFname_tc(String fname_tc) {

        this.fname_tc = fname_tc;

}



public String getLname_tc() {

        return lname_tc;

}



public void setLname_tc(String lname_tc) {

        this.lname_tc = lname_tc;

}



public String getExpertise_tc() {

        return expertise_tc;

}



public void setExpertise_tc(String expertise_tc) {

        this.expertise_tc = expertise_tc;

}
```

```
public String getLevel_tc() {

            return level_tc;

}



public void setLevel_tc(String level_tc) {

            this.level_tc = level_tc;

}






public String getClasses() {

            StringBuilder sb = new StringBuilder();

            if (classes != null && classes.size() > 0) {

                        for (Schoolcls p : classes) {

                                    sb.append(p.getName_cl() );

                        }

}

            return sb.toString();

}



public void setClasses(List<Schoolcls> classes) {

            this.classes = classes;

}



public String getFltrt() {
        return fltrt;
    }



    public void setFltrt(String fltrt) {
        this.fltrt = fltrt;
    }



    @Override
    public String toString() {
        return "Teacher [tid=" + tid + ", fname_tc=" + fname_tc + ", lname_tc=" + lname_tc + ",
expertise_tc="
                + expertise_tc + ", level_tc=" + level_tc + "]";
    }



}

package com.entities;

import java.util.List;

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

import javax.persistence.ManyToMany;

import javax.persistence.Table;


@Entity

@Table (name="Subject_Table")

public class Subject {


        @Id

        @GeneratedValue(strategy = GenerationType.AUTO)

        @Column (name= "Subject_Id")

        private long suid; //subject ID

        @Column (name= "Subject_Name")

        private String name_su; //Subject name

        @Column (name= "Subject_Level")

        private String level_su; //Subject level for Class level

        @Column (name= "Subject_Category")

        private String cat_su; //Subject Category

        @Column(name="Filter_Sub")

        private String fltr_su;



        @ManyToMany(mappedBy= "subjects")

        private List<Schoolcls> classes;




        public Subject() {
        }


        public Subject(String name_su, String level_su, String cat_su, String fltr_su) {


                this.name_su = name_su;
                this.level_su = level_su;
                this.cat_su = cat_su;
                this.fltr_su= fltr_su;
        }




        public long getSuid() {
                return suid;
        }


        public void setSuid(long subjId1) {
                this.suid = subjId1;
        }


        public String getName_su() {
                return name_su;
        }
```

```
public void setName_su(String name_su) {
        this.name_su = name_su;
}


public String getLevel_su() {
        return level_su;
}


public void setLevel_su(String level_su) {
        this.level_su = level_su;
}


public String getCat_su() {
        return cat_su;
}


public void setCat_su(String cat_su) {
        this.cat_su = cat_su;
}

public String getClasses() {
        StringBuilder sb = new StringBuilder();
        if (classes != null && classes.size() > 0) {
                for (Schoolcls p : classes) {
                        sb.append(p.getName_cl() );
                }
}
        return sb.toString();
}


public void setClasses(List<Schoolcls> classes) {
        this.classes = classes;
}




public String getFltr_su() {
        return fltr_su;
}


public void setFltr_su(String fltr_su) {
        this.fltr_su = fltr_su;
}


@Override
public String toString() {
        return "Subject [suid=" + suid + ", name_su=" + name_su + ", level_su=" + level_su +
                        ", cat_su=" + cat_su + "]";
}
```

```
}

package com.entities;

import javax.persistence.CascadeType;
import javax.persistence.Column;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;

import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.Table;




@Entity
@Table (name= "Student_Table")
public class Student {

                @Id()
                @GeneratedValue(strategy = GenerationType.AUTO)
                @Column(name = "Student_ID")
                private long sid;
                @Column(name = "First_Name")
                private String fname_std;
```

```
@Column(name = "Last_Name")
private String lname_std;
@Column(name = "Phone_Number")
private long number_std;
@Column(name = "StudentAddress")
private String saddress;
@Column(name = "StudentEmail")
private String semail;
@Column (name="StudentLevel")
private String slevel;
@Column (name="FilterS")
private String fltrs;//filter for assign



@ManyToOne(cascade=CascadeType.ALL)
@JoinColumn(name="Class_Id")
private Schoolcls sch_class;// school class for each student

public Student() {

}



public Student(String fname_std, String lname_std, long number_std, String saddress,
                String semail, String slevel, String fltrs, Schoolcls sch_class) {


        this.fname_std = fname_std;
```

```
    this.lname_std = lname_std;

    this.number_std = number_std;

    this.saddress = saddress;

    this.semail = semail;

    this.slevel=slevel;

    this.fltrs = fltrs;

    this.sch_class=sch_class;


}



public long getSid() {

    return sid;

}





public void setSid(long sid) {

    this.sid = sid;

}



public String getFname_std() {

    return fname_std;

}

public void setFname_std(String fname_std) {

    this.fname_std = fname_std;

}

public String getLname_std() {
        return lname_std;
}
public void setLname_std(String lname_std) {
        this.lname_std = lname_std;
}
public long getNumber_std() {
        return number_std;
}
public void setNumber_std(long number_std) {
        this.number_std = number_std;
}



public String getSaddress() {
        return saddress;
}



public void setSaddress(String saddress) {
        this.saddress = saddress;
}



public String getSemail() {
        return semail;
}
```

```
public void setSemail(String semail) {
        this.semail = semail;
}


public String getSlevel() {
        return slevel;
}


public void setSlevel(String slevel) {
        this.slevel = slevel;
}


public Schoolcls getSch_class() {


        return sch_class;

}


public void setSch_class(Schoolcls sch_class) {
        this.sch_class = sch_class;
}
```

```
        public String getFltrs() {
            return fltrs;
        }


        public void setFltrs(String fltrs) {
            this.fltrs = fltrs;
        }


        @Override
        public String toString() {
            return "Student [ fname_std=" + fname_std + ", lname_std=" + lname_std + ",
number_std="
                    + number_std + ", saddress=" + saddress + ", semail=" + semail +
"]";
        }
}


package com.entities;


import java.util.Set;


import javax.persistence.CascadeType;
import javax.persistence.Column;


import javax.persistence.Entity;
import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.JoinTable;

import javax.persistence.ManyToMany;

import javax.persistence.OneToMany;

import javax.persistence.Table;





//School Class Entity

@Entity

@Table (name= "Class_Table")

public class Schoolcls {



        @Id

        @GeneratedValue(strategy = GenerationType.AUTO)

        @Column(name="Class_Id")

        private long clid;

        @Column (name="Class_Name")

        private String name_cl;

        @Column (name="Class_Level")

        private String level_cl; //Class level corresponding to the student level

        @Column(name= "Class_Type")

        private String type_cl; //Online or In-class or combined
```

```
    @Column (name="Filter")
    private String fltr; //filter for showing assigned objects


    @ManyToMany(cascade = CascadeType.ALL) //Many to May relationship with subjects
    @JoinTable(name = "Class_Subject", joinColumns = { @JoinColumn(name = "Class_id") },
inverseJoinColumns = {
            @JoinColumn(name = "Subject_id") })
    private Set<Subject> subjects;


    @ManyToMany(cascade = CascadeType.ALL) //Many to Many relationship with teachers
    @JoinTable(name = "Class_Teacher", joinColumns = { @JoinColumn(name = "Class_id") },
inverseJoinColumns = {
            @JoinColumn(name = "Teacher_id") })
    private Set<Teacher> teachers;



    @OneToMany(mappedBy = "sch_class", cascade = CascadeType.ALL) //One to Many
relationship with students
    private Set<Student> students;


    public Schoolcls() {

    }

    public Schoolcls( String name_cl, String level_cl, String type_cl , String fltr) {
            this.name_cl = name_cl;
            this.level_cl = level_cl;
            this.type_cl = type_cl;
            this.fltr= fltr;

    }

    public long getClid() {
            return clid;
    }

    public void setClid(long clid) {
            this.clid = clid;
    }

    public String getName_cl() {

            return name_cl;
    }

    public void setName_cl(String name_cl) {
            this.name_cl = name_cl;
    }

    public String getLevel_cl() {
            return level_cl;
    }
```

```
public void setLevel_cl(String level_cl) {
    this.level_cl = level_cl;
}

public String getType_cl() {
    return type_cl;
}

public void setType_cl(String type_cl) {
    this.type_cl = type_cl;
}

public String getSubjects() {
    StringBuilder sb = new StringBuilder();
    if (subjects != null && subjects.size() > 0) {
        for (Subject p : subjects) {
            sb.append(p.getName_su() );

    }

}
    return sb.toString();
}

public void setSubjects(Set<Subject> subjects) {
    this.subjects = subjects;
}
```

```
public String getTeachers() {


        StringBuilder sb = new StringBuilder();
        if (teachers != null && teachers.size() > 0) {
                for (Teacher p : teachers) {
                        sb.append(p.getFname_tc()+ " " +p.getLname_tc() );
                }
}
        return sb.toString();
}

public void setTeachers(Set<Teacher> teachers) {
        this.teachers = teachers;
}

public String getStudents() {


        StringBuilder sb = new StringBuilder();
        if (students != null && students.size() > 0) {
                for (Student p : students) {
                        sb.append(p.getFname_std()+" "+ p.getLname_std());
                }
}
        return sb.toString();
}

public void setStudents(Set<Student> students) {
        this.students = students;
}
```

```
    public String getFltr() {
        return fltr;
    }

    public void setFltr(String fltr) {
        this.fltr = fltr;
    }
}
```