Learner’s academy

Source Code

Version History

|  |  |
| --- | --- |
| Author | Vivek Ranjan |
| Purpose | Source Code |
| Date | 03/10/2021 |
| Version | 1.0 |

Table of Contents

[Project Code: 2](#_Toc84199236)

[Web.xml 16](#_Toc84199237)

[hibernate.cfg.xml 18](#_Toc84199238)

[hibernateDropTablesFirst.cfg.xml 19](#_Toc84199239)

[HibernateUtil 20](#_Toc84199240)

[Servlet Files 21](#_Toc84199241)

[Com.dao 34](#_Toc84199242)

[Com.dto 45](#_Toc84199243)

# Project Code:

|  |
| --- |
| **Folder Structure** |
|  |
| 1. **Login.jsp** |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>  <%@ include file = *"LoginHeader.jsp"*%>  <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Learner's Academy</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <section class=*"vh-100"* style="background-color: *#508bfc*;">  <div class=*"container py-5 h-100"*;style="overflow: *auto*;display: *inline-block*;">  <div class=*"row d-flex justify-content-center align-items-center h-100"*>  <div class=*"col-12 col-md-8 col-lg-6 col-xl-5"*>  <div class=*"card shadow-2-strong"* style="border-radius: *1rem*;">  <div class=*"card-body p-5 text-center"*>  <h3 class=*"mb-5"*>Sign in</h3>    <form action=*"*<%=request.getContextPath()%>*/login"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"uname"*></label> <input type=*"text"*  class=*"form-control"* id=*"username"* placeholder=*"User Name"*  name=*"uname"* required>  </div>  <div class=*"form-group"*>  <label for=*"uname"*></label> <input type=*"password"*  class=*"form-control"* id=*"password"* placeholder=*"Password"*  name=*"password"* required>  </div>  <button class=*"btn btn-primary btn-lg btn-block"* type=*"submit"*>Login</button>  <div>  <p class=*"mb-0"*>Don't have an account? <a href=*"*<%=request.getContextPath()%>*/signup.jsp"* class=*"text-blue-50 fw-bold"*>Sign Up</a></p>  </div>  </form>  </div>  </section>  </body>  </html> |
| **signup.jsp** |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>  <%@ include file = *"LoginHeader.jsp"*%>  <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Learner's Academy</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <section class=*"vh-100"* style="background-color: *#508bfc*;">  <div class=*"container py-5 h-100"*;style="overflow: *auto*;display: *inline-block*;">  <div class=*"row d-flex justify-content-center align-items-center h-100"*>  <div class=*"col-12 col-md-8 col-lg-6 col-xl-5"*>  <div class=*"card shadow-2-strong"* style="border-radius: *1rem*;">  <div class=*"card-body p-5 text-center"*>  <h3 class=*"mb-5"*>Sign up</h3>    <form action=*"*<%=request.getContextPath()%>*/Signup"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"uname"*></label> <input type=*"text"*  class=*"form-control"* id=*"username"* placeholder=*"User Name"*  name=*"uname"* required>  </div>  <div class=*"form-group"*>  <label for=*"uname"*></label> <input type=*"password"*  class=*"form-control"* id=*"password"* placeholder=*"Password"*  name=*"password"* required>  </div>  <button class=*"btn btn-primary btn-lg btn-block"* type=*"submit"*>Register</button>    </form>  </div>  </section>  </body>  </html> |
| Home.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Class1"*%>  <%@page import=*"com.dto.Student"*%>  <%@page import=*"com.dto.Subject"*%>  <%@ include file = *"Header.jsp"*%>    <% ArrayList class1list = (ArrayList)request.getAttribute("class1list");  ArrayList subjectlist = (ArrayList)request.getAttribute("subjectlist");  ArrayList studentlist = (ArrayList)request.getAttribute("studentlist");  %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Classes</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container-fluid"*>  <div class=*"row"*>  <div class=*"col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"col-md-4"* style="overflow: *auto*;display: *inline-block*;">  <h3 >Subjects and Teacher details</h3>  <table class=*"table"* >  <tr>  <th class=*"col-md-6"*>Subject name</th>  <th class=*"col-md-6"*>Teacher</th>  </tr>  <%**for**(**int** i=0; i<subjectlist.size();i++){ %>  <% Subject subject =(Subject)subjectlist.get(i); %>  <tr class=*'clickable-row'*>  <td>  <%out.print(subject.getSubjectName()); %>  </td>  <td>  <%  **try**{  out.print(subject.getTeacher().getTeacherName());  }**catch**(Exception ex){    }  %>  </td>    </tr>  <% } %>  </table>  </div>  <div class=*"col-md-4 "* style="overflow: *auto*;display: *inline-block*;">  <h3>Student details:</h3>  <table class=*"table"*>  <tr>  <th class=*"col-md-6"*>First name</th>  <th class=*"col-md-6"*>Last name</th>  </tr>  <%**for**(**int** i=0; i<studentlist.size();i++){ %>  <% Student student =(Student)studentlist.get(i); %>  <tr class=*'clickable-row'*>  <td>  <%out.print(student.getStudentFirstName()); %>  </td>  <td>  <% out.print(student.getStudentLastName()); %>  </td>    </tr>  <% } %>  </table>  </div>  <div class=*"col-md-2"* style="overflow: *auto*;display: *inline-block*">  <h3>Search by Class</h3>  <form action=*"*<%=request.getContextPath()%>*/dashboard"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>Class: </label>  <select name=*"classId"* class=*"form-control"* id=*"classId"*>    <% **for**(**int** i=0; i<class1list.size();i++ ){  Class1 class1 =(Class1)class1list.get(i);  %>    <option value=*"*<%out.print(class1.getId()); %>*"*><% out.print(class1.getClassName()); %></option>  <% }%>  </select>  </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"searchByClass"* value=*"searchByClass"*>Search</button>  </form>    </div>  </div>  </div>  </body>  </html> |

|  |
| --- |
| Students.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Student"*%>  <%@page import=*"com.dto.Class1"*%>  <%@ include file = *"Header.jsp"*%>    <% ArrayList list = (ArrayList<Student>)request.getAttribute("studentlist");  ArrayList class1list = (ArrayList)request.getAttribute("classlist");  %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Students</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"container col-md-6"* style="overflow: *auto*;display: *inline-block*;">  <h3>Student Details:</h3>  <table>  <tr>  <th class=*"col-md-6"*>First name</th>  <th class=*"col-md-6"*>Last name</th>  <th class=*"container col-md-6"*>Description</th>  </tr>  <%**for**(**int** i=0; i<list.size();i++){%>  <% Student student =(Student)list.get(i); %>  <tr>  <td >  <% out.print(student.getStudentFirstName()); %>  </td>  <td >  <% out.print(student.getStudentLastName()); %>  </td>  <td>  <% out.print(student.getClass1().getClassName()); %>  </td>  <td><form action=*"*<%=request.getContextPath()%>*/liststudents"* method=*"post"*>  <input style="display:*none*;" type=*"text"* name=*"stId"* value=*"*<%out.print(student.getId());%>*"*>  <button type=*"submit"* class=*"btn"* name=*"deleteStudent"* value=*"deleteStudent"*><i class=*"fa fa-trash"*></i></button>  </form>  </td>  </tr>  <%}%>  </table>  </div>  <div class=*"container col-md-3"* style="overflow: *auto*;display: *inline-block*; position: *absolute*">  <h3>Add New Student</h3>  <form action=*"*<%=request.getContextPath()%>*/liststudents"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>First Name: </label> <input type=*"text"*  class=*"form-control"* id=*"fname"* placeholder=*"First Name"*  name=*"fname"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Last Name: </label> <input type=*"text"*  class=*"form-control"* id=*"lname"* placeholder=*"Last Name"*  name=*"lname"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Class: </label>  <select name=*"class1"* class=*"form-control"* id=*"class1"*>  <!-- option value="0"> select class </option-->  <% **for**(**int** i=0; i<class1list.size();i++ ){  Class1 class1 =(Class1)class1list.get(i);  %>    <option value=*"*<%out.print(class1.getId()); %>*"*><% out.print(class1.getClassName()); %></option>  <% }%>    </select>    </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"addStudent"* value=*"addStudent"*>Add</button>  </form>  </div>  </body>  </html> |
| Classes.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Class1"*%>  <%@ include file = *"Header.jsp"*%>    <% ArrayList classlist = (ArrayList)request.getAttribute("classlist");  %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Classes</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"container col-md-6"* style="overflow: *auto*;display: *inline-block*;">  <h3>Available Class Details:</h3>  <table >  <tr>  <th class=*"col-md-6"*>Class name</th>  <th class=*"container col-md-6"*>Description</th>  <th></th>  </tr>  <%**for**(**int** i=0; i<classlist.size();i++){ %>  <% Class1 class1 =(Class1)classlist.get(i); %>  <tr class=*'clickable-row'*>  <td>  <%out.print(class1.getClassName()); %>  </td>  <td>  <% out.print(class1.getDescription()); %>  </td>  <td><form action=*"*<%=request.getContextPath()%>*/listclasses"* method=*"post"*>  <input style="display:*none*;" type=*"text"* name=*"classId"* value=*"*<%out.print(class1.getId());%>*"*>  <button type=*"submit"* class=*"btn"* name=*"deleteClass"* value=*"deleteClass"*><i class=*"fa fa-trash"*></i></button>  </form>  </td>    </tr>  <% } %>  </table>  </div>  <div class=*"container col-md-3"* style="overflow: *auto*;display: *inline-block*; position: *absolute*">  <h3>Add New Class</h3>  <form action=*"*<%=request.getContextPath()%>*/listclasses"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>Class Name: </label> <input type=*"text"*  class=*"form-control"* id=*"name"* placeholder=*"Class Name"*  name=*"name"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Description: </label> <input type=*"text"*  class=*"form-control"* id=*"discription"* placeholder=*"Description"*  name=*"discription"*>  </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"addClass"* value=*"addClass"*>Add</button>  </form>    </div>    </body>  </html> |
| Header.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Learner's Academy</title>  <style><%@include file=*"/WEB-INF/lib/css/style.css"*%></style>  <link rel=*"stylesheet"* href=*"https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"*>  </head>  <body>  <div class=*"header"*>  <a href=*"#default"* class=*"logo "*>Learner's Academy</a>  <div class=*"header-right"*>  <a id=*"listclasses"* href=*"*<%=request.getContextPath()%>*/dashboard"*>Home</a>  <a id=*"listclasses"* href=*"*<%=request.getContextPath()%>*/listclasses"* >Classes</a>  <a id=*"liststudents"* href=*"*<%=request.getContextPath()%>*/liststudents"* >Students</a>  <a id=*"listteachers"* href=*"*<%=request.getContextPath()%>*/listteachers"* >Teachers</a>  <a id=*"listsubjects"* href=*"*<%=request.getContextPath()%>*/listsubjects"* >Subjects</a>  <a id=*"listlogout"* href=*"*<%=request.getContextPath()%>*/Logout"* >Logout</a>  </div>  </div>  </body>  </html> |
| LoginHeader.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Learner's Academy</title>  <style><%@include file=*"/WEB-INF/lib/css/style.css"*%></style>  </head>  <body>  <div class=*"header"*>  <a href=*"#default"* class=*"logo"*>Learner's Academy</a>  <div class=*"header-right"*>  </div>  </div>  </body>  </html> |
| Students.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Student"*%>  <%@page import=*"com.dto.Class1"*%>  <%@ include file = *"Header.jsp"*%>    <% ArrayList list = (ArrayList<Student>)request.getAttribute("studentlist");  ArrayList class1list = (ArrayList)request.getAttribute("classlist");  %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Students</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"container col-md-6"* style="overflow: *auto*;display: *inline-block*;">  <h3>Student Details:</h3>  <table>  <tr>  <th class=*"col-md-6"*>First name</th>  <th class=*"col-md-6"*>Last name</th>  <th class=*"container col-md-6"*>Description</th>  </tr>  <%**for**(**int** i=0; i<list.size();i++){%>  <% Student student =(Student)list.get(i); %>  <tr>  <td >  <% out.print(student.getStudentFirstName()); %>  </td>  <td >  <% out.print(student.getStudentLastName()); %>  </td>  <td>  <% out.print(student.getClass1().getClassName()); %>  </td>  <td><form action=*"*<%=request.getContextPath()%>*/liststudents"* method=*"post"*>  <input style="display:*none*;" type=*"text"* name=*"stId"* value=*"*<%out.print(student.getId());%>*"*>  <button type=*"submit"* class=*"btn"* name=*"deleteStudent"* value=*"deleteStudent"*><i class=*"fa fa-trash"*></i></button>  </form>  </td>  </tr>  <%}%>  </table>  </div>  <div class=*"container col-md-3"* style="overflow: *auto*;display: *inline-block*; position: *absolute*">  <h3>Add New Student</h3>  <form action=*"*<%=request.getContextPath()%>*/liststudents"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>First Name: </label> <input type=*"text"*  class=*"form-control"* id=*"fname"* placeholder=*"First Name"*  name=*"fname"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Last Name: </label> <input type=*"text"*  class=*"form-control"* id=*"lname"* placeholder=*"Last Name"*  name=*"lname"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Class: </label>  <select name=*"class1"* class=*"form-control"* id=*"class1"*>  <!-- option value="0"> select class </option-->  <% **for**(**int** i=0; i<class1list.size();i++ ){  Class1 class1 =(Class1)class1list.get(i);  %>    <option value=*"*<%out.print(class1.getId()); %>*"*><% out.print(class1.getClassName()); %></option>  <% }%>    </select>    </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"addStudent"* value=*"addStudent"*>Add</button>  </form>  </div>  </body>  </html> |
| Subjects.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Subject"*%>  <%@page import=*"com.dto.Teacher"*%>  <%@page import=*"com.dto.Class1"*%>    <%@ include file = *"Header.jsp"*%>    <% ArrayList subjectlist = (ArrayList)request.getAttribute("subjectlist");  ArrayList class1list = (ArrayList)request.getAttribute("class1list");  ArrayList teacherlist = (ArrayList)request.getAttribute("teacherlist");    %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Subjects</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"container col-md-6"* style="overflow: *auto*;display: *inline-block*;">  <h3>Subjects List</h3>  <table >  <tr>  <th class=*"col-md-3"*>Subject name</th>  <th class=*"col-md-3"*>Class</th>  <th class=*"col-md-3"*>Teacher</th>  <th class=*"col-md-3"*>Description</th>  <th class=*"col-md-2"*></th>  </tr>  <%**for**(**int** i=0; i<subjectlist.size();i++){ %>  <% Subject subject =(Subject)subjectlist.get(i); %>  <tr class=*'clickable-row'*>  <td>  <%out.print(subject.getSubjectName()); %>  </td>  <td>  <%out.print(subject.getClass1().getClassName()); %>  </td>  <td>  <%  **try**{  out.print(subject.getTeacher().getTeacherName());  }**catch**(Exception ex){    }  %>  </td>  <td>  <% out.print(subject.getDescription()); %>  </td>  <td><form action=*"*<%=request.getContextPath()%>*/listsubjects"* method=*"post"*>  <input style="display:*none*;" type=*"text"* name=*"SubjectId"* value=*"*<%out.print(subject.getId());%>*"*>  <button type=*"submit"* class=*"btn"* name=*"deleteSubject"* value=*"deleteSubject"*><i class=*"fa fa-trash"*></i></button>  </form>  </td>    </tr>  <% } %>  </table>  </div>  <div class=*"container col-md-3"* style="overflow: *auto*;display: *inline-block*; position: *absolute*">  <h3>Add New Subject</h3>  <form action=*"*<%=request.getContextPath()%>*/listsubjects"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>Subject Name: </label> <input type=*"text"*  class=*"form-control"* id=*"name"* placeholder=*"Subject Name"*  name=*"name"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Class: </label>  <select name=*"class1"* class=*"form-control"* id=*"class1"*>  <% **for**(**int** i=0; i<class1list.size();i++ ){  Class1 class1 =(Class1)class1list.get(i);  %>    <option value=*"*<%out.print(class1.getId()); %>*"*><% out.print(class1.getClassName()); %></option>  <% }%>    </select>    </div>  <div class=*"form-group"*>  <label for=*"name"*>Teacher: </label>  <select name=*"teacher"* class=*"form-control"* id=*"teacher"*>  <% **for**(**int** i=0; i<teacherlist.size();i++ ){  Teacher teacher =(Teacher)teacherlist.get(i);  %>    <option value=*"*<%out.print(teacher.getId()); %>*"*><% out.print(teacher.getTeacherName()); %></option>  <% }%>    </select>    </div>  <div class=*"form-group"*>  <label for=*"name"*>Description: </label> <input type=*"text"*  class=*"form-control"* id=*"discription"* placeholder=*"Description"*  name=*"discription"*>  </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"addSubject"* value=*"addSubject"*>Add</button>  </form>    </div>    </body>  </html> |
| Teachers.jsp |
| <%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*  pageEncoding=*"ISO-8859-1"*%>    <%@page import=*"java.util.ArrayList"*%>  <%@page import=*"com.dto.Teacher"*%>  <%@ include file = *"Header.jsp"*%>    <% ArrayList teacherlist = (ArrayList)request.getAttribute("teacherlist");  %>      <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Teachers</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <div class=*"container col-md-1"* style="overflow: *auto*;display: *inline-block*;">  </div>  <div class=*"container col-md-6"* style="overflow: *auto*;display: *inline-block*;">  <h3>Teachers List</h3>  <table >  <tr>  <th class=*"col-md-6"*>Teacher name</th>  <th class=*"container col-md-6"*>Description</th>  <th></th>  </tr>  <%**for**(**int** i=0; i<teacherlist.size();i++){ %>  <% Teacher teacher =(Teacher)teacherlist.get(i); %>  <tr class=*'clickable-row'*>  <td>  <%out.print(teacher.getTeacherName()); %>  </td>  <td>  <% out.print(teacher.getDescription()); %>  </td>  <td><form action=*"*<%=request.getContextPath()%>*/listteachers"* method=*"post"*>  <input style="display:*none*;" type=*"text"* name=*"TeacherId"* value=*"*<%out.print(teacher.getId());%>*"*>  <button type=*"submit"* class=*"btn"* name=*"deleteTeacher"* value=*"deleteTeacher"*><i class=*"fa fa-trash"*></i></button>  </form>  </td>    </tr>  <% } %>  </table>  </div>  <div class=*"container col-md-3"* style="overflow: *auto*;display: *inline-block*; position: *absolute*">  <h3>Add New Teacher</h3>  <form action=*"*<%=request.getContextPath()%>*/listteachers"* method=*"post"*>  <div class=*"form-group"*>  <label for=*"name"*>Teacher Name: </label> <input type=*"text"*  class=*"form-control"* id=*"name"* placeholder=*"Teacher Name"*  name=*"name"* required>  </div>  <div class=*"form-group"*>  <label for=*"name"*>Description: </label> <input type=*"text"*  class=*"form-control"* id=*"discription"* placeholder=*"Description"*  name=*"discription"*>  </div>  <button type=*"submit"* class=*"btn btn-primary"* name=*"addTeacher"* value=*"addTeacher"*>Add</button>  </form>    </div>    </body>  </html> |
| Logout.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>Logout</title>  <link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"*>  </head>  <body>  <%="Thank you!" %>  </body>  </html> |

# Web.xml

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"* id=*"WebApp\_ID"* version=*"2.5"*>  <display-name>Learners\_Academy</display-name>  <welcome-file-list>  <welcome-file>index.html</welcome-file>  <welcome-file>index.htm</welcome-file>  <welcome-file>index.jsp</welcome-file>  <welcome-file>default.html</welcome-file>  <welcome-file>default.htm</welcome-file>  <welcome-file>default.jsp</welcome-file>  </welcome-file-list>  <servlet>  <description></description>  <display-name>Login</display-name>  <servlet-name>Login</servlet-name>  <servlet-class>com.controllers.Login</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>Login</servlet-name>  <url-pattern>/login</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>Login2</display-name>  <servlet-name>Login2</servlet-name>  <servlet-class>com.controllers.Login2</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>Login2</servlet-name>  <url-pattern>/Login2</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>ListClasses</display-name>  <servlet-name>ListClasses</servlet-name>  <servlet-class>com.controllers.ListClasses</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>ListClasses</servlet-name>  <url-pattern>/listclasses</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>ListStudents</display-name>  <servlet-name>ListStudents</servlet-name>  <servlet-class>com.controllers.ListStudents</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>ListStudents</servlet-name>  <url-pattern>/liststudents</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>ListTeachers</display-name>  <servlet-name>ListTeachers</servlet-name>  <servlet-class>com.controllers.ListTeachers</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>ListTeachers</servlet-name>  <url-pattern>/listteachers</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>ListSubjects</display-name>  <servlet-name>ListSubjects</servlet-name>  <servlet-class>com.controllers.ListSubjects</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>ListSubjects</servlet-name>  <url-pattern>/listsubjects</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>Dashboard</display-name>  <servlet-name>Dashboard</servlet-name>  <servlet-class>com.controllers.Dashboard</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>Dashboard</servlet-name>  <url-pattern>/dashboard</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>Logout</display-name>  <servlet-name>Logout</servlet-name>  <servlet-class>com.controllers.Logout</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>Logout</servlet-name>  <url-pattern>/logout</url-pattern>  </servlet-mapping>  <servlet>  <description></description>  <display-name>Signup</display-name>  <servlet-name>Signup</servlet-name>  <servlet-class>com.controllers.Signup</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>Signup</servlet-name>  <url-pattern>/signup</url-pattern>  </servlet-mapping>  </web-app> |

# hibernate.cfg.xml

|  |
| --- |
|  |
| <?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>  <!-- >property name="connection.driver\_class">com.mysql.cj.jdbc.Driver</property -->  <property name=*"connection.driver\_class"*>com.mysql.jdbc.Driver</property>  <property name=*"connection.url"*>jdbc:mysql://localhost:3306/learnersAcademy</property>  <property name=*"connection.username"*>root</property>  <property name=*"connection.password"*>W3lcome1234!</property>    <property name=*"connection.pool\_size"*>3</property>    <property name=*"dialect"*>org.hibernate.dialect.MySQL8Dialect</property>  <property name=*"current\_session\_context\_class"*>thread</property>    <property name=*"show\_sql"*>true</property>  <property name=*"format\_sql"*>true</property>    <property name=*"hbm2ddl.auto"*>update</property>    <mapping class=*"com.dto.Subject"*/>  <mapping class=*"com.dto.Class1"*/>  <mapping class=*"com.dto.Student"*/>    <mapping class=*"com.dto.Teacher"*/>  <mapping class=*"com.dto.User"*/>    </session-factory>  </hibernate-configuration> |

|  |
| --- |
| hibernateDropTablesFirst.cfg.xml |
| <?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>  <!-- >property name="connection.driver\_class">com.mysql.cj.jdbc.Driver</property -->  <property name=*"connection.driver\_class"*>com.mysql.jdbc.Driver</property>  <property name=*"connection.url"*>jdbc:mysql://localhost:3306/learnersAcademy</property>  <property name=*"connection.username"*>root</property>  <property name=*"connection.password"*>root</property>    <property name=*"connection.pool\_size"*>3</property>    <property name=*"dialect"*>org.hibernate.dialect.MySQL8Dialect</property>  <property name=*"current\_session\_context\_class"*>thread</property>    <property name=*"show\_sql"*>true</property>  <property name=*"format\_sql"*>true</property>    <property name=*"hbm2ddl.auto"*>update</property>    <mapping class=*"com.dto.Class1"*/>  <mapping class=*"com.dto.Student"*/>  <mapping class=*"com.dto.Subject"*/>  <mapping class=*"com.dto.Teacher"*/>  <mapping class=*"com.dto.User"*/>    </session-factory>  </hibernate-configuration> |

|  |
| --- |
| HibernateUtil |
| package com.util;  import org.hibernate.SessionFactory;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  public class HibernateUtil {  private static StandardServiceRegistry registry;  private static SessionFactory sessionFactory;  public static SessionFactory getSessionFactory() {  if (sessionFactory == null) {  try {  // Create registry  registry = new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  // Create MetadataSources  MetadataSources sources = new MetadataSources(registry);  // Create Metadata  Metadata metadata = sources.getMetadataBuilder().build();  // Create SessionFactory  sessionFactory = metadata.getSessionFactoryBuilder().build();  } catch (Exception e) {  e.printStackTrace();  if (registry != null) {  StandardServiceRegistryBuilder.destroy(registry);  }  }  }  return sessionFactory;  }  public static void shutdown() {  if (registry != null) {  StandardServiceRegistryBuilder.destroy(registry);  }  }  } |

# Servlet Files

|  |
| --- |
| Dashboard.java |
| package com.controllers;  import java.io.IOException;  import java.util.ArrayList;  import java.util.List;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.Class1DoaImp;  import com.dao.StudentDaoImp;  import com.dao.SubjectDaoImp;  import com.dto.Class1;  import com.dto.Student;  import com.dto.Subject;  /\*\*  \* Servlet implementation class Dashboard  \*/  public class Dashboard extends HttpServlet {  private static final long serialVersionUID = 1L;  /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public Dashboard() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse  \* response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  HttpSession session = request.getSession();  String loggeduname = (String) session.getAttribute("uname");  if (loggeduname != null) {  Class1DoaImp class1DoaImp = new Class1DoaImp();  SubjectDaoImp subjectDaoImp = new SubjectDaoImp();  StudentDaoImp studentDaoImp = new StudentDaoImp();  Integer classId = 0;  if (request.getParameter("If-Modified-Since") == null && request.getParameter("searchByClass") != null  && request.getParameter("searchByClass").equals("searchByClass")  && request.getParameter("classId") != null) {  System.out.println(request.getParameter("classId"));  classId = Integer.parseInt(request.getParameter("classId"));  }  List<Class1> class1list = class1DoaImp.listAllClasses();  ArrayList<Subject> subjectlist = (ArrayList<Subject>) subjectDaoImp.searchSubjectsByClassId(classId);  List<Student> studentlist = studentDaoImp.searchStudentsByClassId(classId);  request.setAttribute("subjectlist", subjectlist);  request.setAttribute("class1list", class1list);  request.setAttribute("studentlist", studentlist);  RequestDispatcher rd = request.getRequestDispatcher("Home.jsp");  rd.forward(request, response);  } else {  RequestDispatcher rd = request.getRequestDispatcher("login");  request.setAttribute("loginstatus", "Logged out");  rd.forward(request, response);  }  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse  \* response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |
| ListClasses.java |
| package com.controllers;  import java.io.IOException;  import java.util.ArrayList;  //import java.util.List;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.Class1DoaImp;  import com.dto.Class1;  /\*\*  \* Servlet implementation class ListClasses  \*/  public class ListClasses extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public ListClasses() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub    HttpSession session = request.getSession();  String loggeduname= (String)session.getAttribute("uname");  if(loggeduname!=null){  Class1DoaImp class1DoaImp= new Class1DoaImp();    if (request.getParameter("If-Modified-Since") == null && request.getParameter("addClass")!=null && request.getParameter("addClass").equals("addClass") && request.getParameter("name")!=null)  {  System.out.println("test form");  System.out.println(request.getParameter("addClass"));  System.out.println(request.getParameter("className"));  System.out.println("test form end");    Class1 NewClass = new Class1(request.getParameter("name"),request.getParameter("discription"));  class1DoaImp.addClass(NewClass);    }  if (request.getParameter("If-Modified-Since") == null && request.getParameter("deleteClass")!=null && request.getParameter("deleteClass").equals("deleteClass") && request.getParameter("classId")!=null)  {  System.out.println(request.getParameter("classId"));  Integer classId = Integer.parseInt(request.getParameter("classId"));  class1DoaImp.deleteClass1(classId);    }        ArrayList<Class1> classlist=(ArrayList<Class1>) class1DoaImp.listAllClasses();    request.setAttribute("classlist", classlist);    RequestDispatcher rd = request.getRequestDispatcher("Classes.jsp");      rd.forward(request, response);    }  else {  RequestDispatcher rd = request.getRequestDispatcher("login");  request.setAttribute("loginstatus", "Logged out");  rd.forward(request, response);  }      }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);    }  } |

|  |
| --- |
| ListStudents.java |
| package com.controllers;  import java.io.IOException;  import java.util.List;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.Class1DoaImp;  import com.dao.StudentDaoImp;  import com.dto.Class1;  import com.dto.Student;  /\*\*  \* Servlet implementation class ListStudents  \*/  public class ListStudents extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public ListStudents() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {    HttpSession session = request.getSession();  String loggeduname= (String)session.getAttribute("uname");  if(loggeduname!=null){  Class1DoaImp class1DoaImp= new Class1DoaImp();  StudentDaoImp studentDaoImp= new StudentDaoImp();  System.out.println(request.getParameter("If-Modified-Since"));    if (request.getParameter("If-Modified-Since") == null && request.getParameter("addStudent")!=null && request.getParameter("addStudent").equals("addStudent") && request.getParameter("fname")!=null)  {    Class1 class1 = class1DoaImp.searchClass1ById(Integer.valueOf(request.getParameter("class1")));    Student NewStudent = new Student(request.getParameter("fname"),request.getParameter("lname"),class1);  studentDaoImp.addStudent(NewStudent);    }    if (request.getParameter("If-Modified-Since") == null && request.getParameter("deleteStudent")!=null && request.getParameter("deleteStudent").equals("deleteStudent") && request.getParameter("stId")!=null)  {    int stId = Integer.parseInt(request.getParameter("stId"));  studentDaoImp.deleteStudent(stId);    }        List<Class1> classlist=class1DoaImp.listAllClasses();  List<Student> students=studentDaoImp.listAllStudents();    request.setAttribute("classlist", classlist);  request.setAttribute("studentlist", students);    RequestDispatcher rd = request.getRequestDispatcher("Students.jsp");      rd.forward(request, response);  }  else {  RequestDispatcher rd = request.getRequestDispatcher("login");  request.setAttribute("loginstatus", "Logged out");  rd.forward(request, response);  }  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

|  |
| --- |
| ListSubjects.java |
| package com.controllers;  import java.io.IOException;  import java.util.ArrayList;  import java.util.List;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.Class1DoaImp;  import com.dao.SubjectDaoImp;  import com.dao.TeacherDaoImp;  import com.dto.Class1;  import com.dto.Subject;  import com.dto.Teacher;  /\*\*  \* Servlet implementation class ListSubjects  \*/  public class ListSubjects extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public ListSubjects() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  HttpSession session = request.getSession();  String loggeduname= (String)session.getAttribute("uname");  if(loggeduname!=null){  Class1DoaImp class1DoaImp= new Class1DoaImp();  TeacherDaoImp teacherDaoImp= new TeacherDaoImp();  SubjectDaoImp subjectDaoImp= new SubjectDaoImp();    if (request.getParameter("If-Modified-Since") == null && request.getParameter("addSubject")!=null && request.getParameter("addSubject").equals("addSubject") && request.getParameter("name")!=null)  {  Class1 class1 = class1DoaImp.searchClass1ById(Integer.valueOf(request.getParameter("class1")));  Teacher teacher= teacherDaoImp.searchTeacherById(Integer.valueOf(request.getParameter("teacher")));  Subject obj = new Subject(request.getParameter("name"),request.getParameter("discription"),class1,teacher);  subjectDaoImp.addSubject(obj);      }  if (request.getParameter("If-Modified-Since") == null && request.getParameter("deleteSubject")!=null && request.getParameter("deleteSubject").equals("deleteSubject") && request.getParameter("SubjectId")!=null)  {  Integer SubjectId = Integer.parseInt(request.getParameter("SubjectId"));  subjectDaoImp.deleteSubject(SubjectId);    }      List<Class1> class1list=class1DoaImp.listAllClasses();  ArrayList<Subject> subjectlist=(ArrayList<Subject>) subjectDaoImp.listAllSubjects();  List<Teacher> teacherlist=teacherDaoImp.listAllTeachers();    request.setAttribute("subjectlist", subjectlist);  request.setAttribute("class1list", class1list);  request.setAttribute("teacherlist", teacherlist);      RequestDispatcher rd = request.getRequestDispatcher("Subjects.jsp");      rd.forward(request, response);    }  else {  RequestDispatcher rd = request.getRequestDispatcher("login");  request.setAttribute("loginstatus", "Logged out");  rd.forward(request, response);  }  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

|  |
| --- |
| ListTeachers.java |
| package com.controllers;  import java.io.IOException;  import java.util.ArrayList;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.TeacherDaoImp;  import com.dto.Teacher;  /\*\*  \* Servlet implementation class ListTeachers  \*/  public class ListTeachers extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public ListTeachers() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  HttpSession session = request.getSession();  String loggeduname= (String)session.getAttribute("uname");  if(loggeduname!=null){  TeacherDaoImp teacherDaoImp= new TeacherDaoImp();    if (request.getParameter("If-Modified-Since") == null && request.getParameter("addTeacher")!=null && request.getParameter("addTeacher").equals("addTeacher") && request.getParameter("name")!=null)  {  Teacher obj = new Teacher(request.getParameter("name"),request.getParameter("discription"));  teacherDaoImp.addTeacher(obj);    }  if (request.getParameter("If-Modified-Since") == null && request.getParameter("deleteTeacher")!=null && request.getParameter("deleteTeacher").equals("deleteTeacher") && request.getParameter("TeacherId")!=null)  {  Integer TeacherId = Integer.parseInt(request.getParameter("TeacherId"));  teacherDaoImp.deleteTeacher(TeacherId);    }        ArrayList<Teacher> teacherlist=(ArrayList<Teacher>) teacherDaoImp.listAllTeachers();    request.setAttribute("teacherlist", teacherlist);    RequestDispatcher rd = request.getRequestDispatcher("Teachers.jsp");      rd.forward(request, response);    }  else {  RequestDispatcher rd = request.getRequestDispatcher("login");  request.setAttribute("loginstatus", "Logged out");  rd.forward(request, response);  }  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

|  |
| --- |
| Login.java |
| package com.controllers;  import java.io.IOException;  import javax.servlet.ServletConfig;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import javax.servlet.http.HttpSession;  import com.dao.UserDaoImp;  import com.dto.User;  /\*\*  \* Servlet implementation class Login  \*/  public class Login extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public Login() {  super();  // TODO Auto-generated constructor stub  }    public void init(ServletConfig config) throws ServletException {  /\*  \* User user = new User("admin","admin"); UserDaoImp userDaoimp= new  \* UserDaoImp(); userDaoimp.addUser(user);  \*/    }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub    String uname = request.getParameter("uname");  String password = request.getParameter("password");    User user = new User(uname,password);  UserDaoImp dao = new UserDaoImp();  boolean isValiduser = dao.validateUser(user);    if (isValiduser) {  HttpSession session = request.getSession();  session.setAttribute("uname",user.getUserName());  session.setMaxInactiveInterval(600);  response.sendRedirect("dashboard");  }else {  response.sendRedirect("login.jsp");  }  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

|  |
| --- |
| Logout.java |
| package com.controllers;  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 javax.servlet.http.HttpSession;  /\*\*  \* Servlet implementation class Logout  \*/  @WebServlet("/Logout")  public class Logout extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public Logout() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see Servlet#destroy()  \*/  public void destroy() {  // TODO Auto-generated method stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  HttpSession session = request.getSession();  session.invalidate();  PrintWriter out=response.getWriter();  out.print("You are successfully logged out!");    response.sendRedirect("login");  out.close();  }  /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

|  |
| --- |
| Signup.java |
| package com.controllers;  import java.io.IOException;  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 javax.servlet.http.HttpSession;  import com.dao.UserDaoImp;  import com.dto.User;  /\*\*  \* Servlet implementation class Signup  \*/  @WebServlet("/Signup")  public class Signup extends HttpServlet {  private static final long serialVersionUID = 1L;    /\*\*  \* @see HttpServlet#HttpServlet()  \*/  public Signup() {  super();  // TODO Auto-generated constructor stub  }  /\*\*  \* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub    String uname = request.getParameter("uname");    String password = request.getParameter("password");  System.out.println(uname);  System.out.println(password);  User user = new User(uname,password);  UserDaoImp userDaoimp= new UserDaoImp();  userDaoimp.addUser(user);  System.out.println("New user created");  response.sendRedirect("login.jsp");    }    /\*\*  \* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)  \*/  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

# Com.dao

|  |
| --- |
| Class1Dao.java |
| package com.dao;  import java.util.List;  import com.dto.Class1;  public interface Class1Dao {    public List<Class1> listAllClasses();  public Class1 addClass(Class1 class1);  public void updateClass1(Class1 class1);  public void deleteClass1(int classId);  public Class1 searchClass1ById(int id);  } |

|  |
| --- |
| Class1DoaImp.java |
| package com.dao;  import java.util.List;  import javax.persistence.TypedQuery;  import org.hibernate.Session;  import org.hibernate.SessionFactory;  import org.hibernate.Transaction;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  import com.dto.Class1;  public class Class1DoaImp implements Class1Dao {    private SessionFactory factory;    public Class1DoaImp() {  StandardServiceRegistry ssr = new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build();  factory = meta.getSessionFactoryBuilder().build();  }  @SuppressWarnings("unchecked")  @Override  public List<Class1> listAllClasses() {  List<Class1> classes =null;  Session session=factory.openSession();    String hql="from Class1";    TypedQuery<Class1> query= session.createQuery(hql);    classes=query.getResultList();  session.close();  return classes;  }  @Override  public Class1 addClass(Class1 class1) {  Session session = factory.openSession();  Transaction trx = session.beginTransaction();    session.save(class1);    trx.commit();  session.close();  return class1;  }  @Override  public void updateClass1(Class1 class1) {  Session session = factory.openSession();  Transaction trx = session.beginTransaction();    session.update(class1);    trx.commit();  session.close();    }  @Override  public Class1 searchClass1ById(int id) {  Session session = factory.openSession();  //Transaction txn = session.beginTransaction();  String hql = "FROM Class1 e where e.id = "+ id;  @SuppressWarnings("unchecked")  TypedQuery<Class1> query = session.createQuery(hql);  Class1 class1 = query.getSingleResult();  session.close();  return class1;      }  @Override  public void deleteClass1(int classId) {  Session session = factory.openSession();  Transaction txn = session.beginTransaction();  Class1 class1 = session.get(Class1.class, classId);  session.delete(class1);  txn.commit();  session.close();    }        } |
| StudentDao.java |
| package com.dao;  import java.util.List;  import com.dto.Student;  public interface StudentDao {  public List<Student> listAllStudents();  public Student addStudent(Student student);  void deleteStudent(int stId);  public List<Student> searchStudentsByClassId(int classId);  } |

|  |
| --- |
| StudentDaoImp.java |
| package com.dao;  import java.util.List;  import javax.persistence.TypedQuery;  import org.hibernate.Session;  import org.hibernate.SessionFactory;  import org.hibernate.Transaction;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  import com.dto.Student;  public class StudentDaoImp implements StudentDao {    private SessionFactory factory;    public StudentDaoImp() {  StandardServiceRegistry ssr = new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build();  factory = meta.getSessionFactoryBuilder().build();  }  @Override  public List<Student> listAllStudents() {  List<Student> students =null;  Session session=factory.openSession();    String hql="from Student";    @SuppressWarnings("unchecked")  TypedQuery<Student> query= session.createQuery(hql);    students=query.getResultList();  session.close();  return students;  }  @Override  public Student addStudent(Student student) {  Session session = factory.openSession();  Transaction trx = session.beginTransaction();    session.save(student);    trx.commit();  session.close();  return student;  }  @Override  public void deleteStudent(int stId) {  Session session = factory.openSession();  Transaction txn = session.beginTransaction();  Student student = session.get(Student.class, stId);  session.delete(student);  txn.commit();  session.close();      }  @Override  public List<Student> searchStudentsByClassId(int classId) {  List<Student> students = null;  Session session = factory.openSession();  //Transaction txn = session.beginTransaction();  String hql = "FROM Student e where e.class1 = "+ classId;  @SuppressWarnings("unchecked")  TypedQuery<Student> query = session.createQuery(hql);  students = query.getResultList();  session.close();  return students;  }  } |

|  |
| --- |
| SubjectDao.java |
| package com.dao;  import java.util.List;  import com.dto.Subject;  public interface SubjectDao {  public List<Subject> listAllSubjects();  public Subject addSubject(Subject subject);  public void updateSubject(Subject subject);  public void deleteSubject(int subjectId);  public Subject searchSubjectById(int id);  public List<Subject> searchSubjectByTeacherId(int teacherId);  public List<Subject> searchSubjectsByClassId(int classId);  } |

|  |
| --- |
| SubjectDaoImp.java |
| package com.dao;  import java.util.List;  import javax.persistence.TypedQuery;  import org.hibernate.Session;  import org.hibernate.SessionFactory;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  import org.hibernate.Transaction;  import com.dto.Subject;  public class SubjectDaoImp implements SubjectDao {    private SessionFactory factory;    public SubjectDaoImp() {  StandardServiceRegistry ssr = new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build();  factory = meta.getSessionFactoryBuilder().build();  }  @Override  public List<Subject> listAllSubjects() {  List<Subject> subjects =null;  Session session=factory.openSession();    String hql="from Subject";    @SuppressWarnings("unchecked")  TypedQuery<Subject> query= session.createQuery(hql);    subjects=query.getResultList();  session.close();  return subjects;  }  @Override  public Subject addSubject(Subject subject) {  Session session = factory.openSession();  Transaction trx = session.beginTransaction();    session.save(subject);    trx.commit();  session.close();  return subject;  }  @Override  public void updateSubject(Subject subject) {  // TODO Auto-generated method stub    }  @Override  public void deleteSubject(int subjectId) {  Session session = factory.openSession();  Transaction txn = session.beginTransaction();  Subject subject = session.get(Subject.class, subjectId);  session.delete(subject);  txn.commit();  session.close();    }  @Override  public Subject searchSubjectById(int id) {  // TODO Auto-generated method stub  return null;  }  @Override  public List<Subject> searchSubjectByTeacherId(int teacherId) {  List<Subject> subjects = null;  Session session = factory.openSession();  //Transaction txn = session.beginTransaction();  String hql = "FROM Subject e where e.teacher = "+ teacherId;  @SuppressWarnings("unchecked")  TypedQuery<Subject> query = session.createQuery(hql);  subjects = query.getResultList();  session.close();  return subjects;  }  @Override  public List<Subject> searchSubjectsByClassId(int classId) {  List<Subject> subjects = null;  Session session = factory.openSession();  //Transaction txn = session.beginTransaction();  String hql = "FROM Subject e where e.class1 = "+ classId;  @SuppressWarnings("unchecked")  TypedQuery<Subject> query = session.createQuery(hql);  subjects = query.getResultList();  session.close();  return subjects;  }    } |

|  |
| --- |
| TeacherDao |
| package com.dao;  import java.util.List;  import com.dto.Teacher;  public interface TeacherDao {  public List<Teacher> listAllTeachers();  public Teacher addTeacher(Teacher teacher);  public void updateTeacher(Teacher teacher);  public void deleteTeacher(int teacherId);  public Teacher searchTeacherById(int id);  } |

|  |
| --- |
| TeacherDaoImp.java |
| package com.dao;  import java.util.List;  import javax.persistence.TypedQuery;  import org.hibernate.Session;  import org.hibernate.SessionFactory;  import org.hibernate.Transaction;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  import com.dto.Subject;  import com.dto.Teacher;  public class TeacherDaoImp implements TeacherDao {    private SessionFactory factory;    public TeacherDaoImp() {  StandardServiceRegistry ssr = new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build();  factory = meta.getSessionFactoryBuilder().build();  }  @SuppressWarnings("unchecked")  @Override  public List<Teacher> listAllTeachers() {  List<Teacher> teachers =null;  Session session=factory.openSession();    String hql="from Teacher";    TypedQuery<Teacher> query= session.createQuery(hql);    teachers=query.getResultList();    session.close();  return teachers;  }  @Override  public Teacher addTeacher(Teacher teacher) {  Session session = factory.openSession();  Transaction trx = session.beginTransaction();    session.save(teacher);    trx.commit();  session.close();  return teacher;  }  @Override  public void updateTeacher(Teacher teacher) {  // TODO Auto-generated method stub    }  @Override  public void deleteTeacher(int teacherId) {  Session session = factory.openSession();  Transaction txn = session.beginTransaction();  Teacher teacher = session.get(Teacher.class, teacherId);    SubjectDaoImp subjectDoaImpObj= new SubjectDaoImp();  List<Subject> subjects = null;  subjects=subjectDoaImpObj.searchSubjectByTeacherId(teacherId);  System.out.println("Test teacher delete" + subjects.size() +"teacherId: " +teacherId);  for(int i=0;i<subjects.size();i++) {  System.out.println(0);  Subject subjectObj= new Subject();  subjectObj=subjects.get(i);  subjectObj.setTeacher(null);  session.update(subjectObj);  System.out.println(i);  System.out.println(subjectObj.getSubjectName());  }    session.delete(teacher);  txn.commit();  session.close();    }  @Override  public Teacher searchTeacherById(int id) {  Session session = factory.openSession();  //Transaction txn = session.beginTransaction();  String hql = "FROM Teacher e where e.id = "+ id;  @SuppressWarnings("unchecked")  TypedQuery<Teacher> query = session.createQuery(hql);  Teacher teacher = query.getSingleResult();  session.close();  return teacher;  }  } |

|  |
| --- |
| UserDao.java |
| **package** com.dao;  **import** com.dto.User;  **public** **interface** UserDao {    **public** **void** addUser(User user);  **public** **boolean** validateUser(User user);  } |

|  |
| --- |
| UserDaoImp.java |
| package com.dao;  import org.hibernate.Session;  import org.hibernate.SessionFactory;  import org.hibernate.Transaction;  import org.hibernate.boot.Metadata;  import org.hibernate.boot.MetadataSources;  import org.hibernate.boot.registry.StandardServiceRegistry;  import org.hibernate.boot.registry.StandardServiceRegistryBuilder;  import com.dto.User;  import com.util.HibernateUtil;  public class UserDaoImp implements UserDao {    private SessionFactory factory;    /\*  \* public UserDaoImp() { StandardServiceRegistry ssr = new  \* StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();  \* Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build();  \* factory = meta.getSessionFactoryBuilder().build(); }  \*/  @Override  public void addUser (User user) {    Transaction transaction = null;  User validUser = null;  try (Session session = HibernateUtil.getSessionFactory().openSession()) {  // start a transaction  session.beginTransaction();  // get an user object      session.save(user);  session.getTransaction().commit();  session.close();  }catch(Exception e){  System.out.println(e.getMessage());  }  }    @Override  public boolean validateUser(User user) {    Transaction transaction = null;  User validUser = null;  try (Session session = HibernateUtil.getSessionFactory().openSession()) {  // start a transaction  transaction = session.beginTransaction();  // get an user object  validUser = (User) session.createQuery("FROM User U WHERE U.userName = :userName").setParameter("userName", user.getUserName())  .uniqueResult();  if (user != null && validUser.getPassword().equals(user.getPassword())) {  return true;  }  // commit transaction  //transaction.commit();  } catch (Exception e) {  if (transaction != null) {  //transaction.rollback();  }  e.printStackTrace();  }  return false;  }    } |

# Com.dto

|  |
| --- |
| Class1.java |
| package com.dto;  import java.util.List;  import javax.persistence.CascadeType;  import javax.persistence.Column;  import javax.persistence.Entity;  import javax.persistence.FetchType;  import javax.persistence.GeneratedValue;  import javax.persistence.GenerationType;  import javax.persistence.Id;  import javax.persistence.Inheritance;  import javax.persistence.InheritanceType;  import javax.persistence.OneToMany;  import javax.persistence.OrderColumn;  import javax.persistence.Table;  @Entity  @Inheritance(strategy = InheritanceType.TABLE\_PER\_CLASS)  @Table(name="classes")  public class Class1 {  @Id  @GeneratedValue(strategy=GenerationType.TABLE)  private int id;    @Column(name="class\_name")  private String className;    @Column(name="disc")  private String description;    @OneToMany(fetch = FetchType.LAZY, cascade=CascadeType.ALL, mappedBy="class1")  @OrderColumn(name="id")  private List<Subject> subjects;      public String getClassName() {  return className;  }  public void setClassName(String className) {  this.className = className;  }  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getDescription() {  return description;  }  public void setDescription(String description) {  this.description = description;  }    public List<Subject> getSubjects() {  return subjects;  }  public void setSubjects(List<Subject> subjects) {  this.subjects = subjects;  }  public Class1(){    }    public Class1(String className){  this.className=className;  }  public Class1(String className, String description) {  this.className=className;  this.description=description;    }  } |

|  |
| --- |
| Student.java |
| package com.dto;  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;  import javax.persistence.ForeignKey;  @Entity  @Table(name="student")  public class Student {    @Id  @GeneratedValue(strategy=GenerationType.TABLE)  private int id;    @Column(name="student\_fname")  private String studentFirstName;    @Column(name="student\_lname")  private String studentLastName;    @ManyToOne(cascade=CascadeType.DETACH)  @JoinColumn(name="class\_id", foreignKey=@ForeignKey(name="class\_id\_foreign"))  private Class1 class1;      public Student(){    }    public Student(String studentFirstName,String studentLastName){  this.studentFirstName=studentFirstName;  this.studentLastName=studentLastName;  }  public Student(String fname, String lname, Class1 class1) {  // TODO Auto-generated constructor stub  this.studentFirstName=fname;  this.studentLastName=lname;  this.class1=class1;  }  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getStudentFirstName() {  return studentFirstName;  }  public void setStudentFirstName(String studentFirstName) {  this.studentFirstName = studentFirstName;  }  public String getStudentLastName() {  return studentLastName;  }  public void setStudentLastName(String studentLastName) {  this.studentLastName = studentLastName;  }  public Class1 getClass1() {  return class1;  }  public void setClass1(Class1 class1) {  this.class1 = class1;  }      } |

|  |
| --- |
| Subject.java |
| package com.dto;  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="subject")  public class Subject {    @Id  @GeneratedValue(strategy=GenerationType.TABLE)  private int id;  @Column(name="subjectName")  private String subjectName;  @Column(name="disc")  private String description;  @ManyToOne  @JoinColumn(name="classId")  private Class1 class1;  @ManyToOne  @JoinColumn(name="teacherId")  private Teacher teacher;      public Subject(){    }  public Subject(String subjectName,String description, Class1 class1, Teacher teacher){  this.subjectName=subjectName;  this.description=description;  this.class1=class1;  this.teacher=teacher;    }    public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getSubjectName() {  return subjectName;  }  public void setSubjectName(String subjectName) {  this.subjectName = subjectName;  }  public String getDescription() {  return description;  }  public void setDescription(String description) {  this.description = description;  }  public Class1 getClass1() {  return class1;  }  public void setClass1(Class1 class1) {  this.class1 = class1;  }  public Teacher getTeacher() {  return teacher;  }  public void setTeacher(Teacher teacher) {  this.teacher = teacher;  }    } |

|  |
| --- |
| Teacher.java |
| package com.dto;  import java.util.List;  import javax.persistence.Column;  import javax.persistence.Entity;  import javax.persistence.FetchType;  import javax.persistence.GeneratedValue;  import javax.persistence.GenerationType;  import javax.persistence.Id;  import javax.persistence.OneToMany;  import javax.persistence.OrderColumn;  import javax.persistence.Table;  @Entity  @Table(name="teacher")  public class Teacher {    public Teacher(String teacherName, String description) {  this.teacherName=teacherName;  this.description=description;  }      public Teacher() {  super();  }  @Id  @GeneratedValue(strategy=GenerationType.TABLE)  private int id;    @Column(name="teacherName")  private String teacherName;    @Column(name="disc")  private String description;    @OneToMany(fetch = FetchType.LAZY, mappedBy="teacher")  @OrderColumn(name="id")  private List<Subject> subjects;  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getTeacherName() {  return teacherName;  }  public void setTeacherName(String teacherName) {  this.teacherName = teacherName;  }  public String getDescription() {  return description;  }  public void setDescription(String description) {  this.description = description;  }  public List<Subject> getSubjects() {  return subjects;  }  public void setSubjects(List<Subject> subjects) {  this.subjects = subjects;  }      } |

|  |
| --- |
| User.java |
| package com.dto;  import javax.persistence.Column;  import javax.persistence.Entity;  import javax.persistence.Id;  import javax.persistence.Table;  @Entity  @Table(name="user")  public class User {    public User(String uname, String password) {  this.userName=uname;  this.password=password;  }  public User() {  // TODO Auto-generated constructor stub  }  public String getUserName() {  return userName;  }  public void setUserName(String userName) {  this.userName = userName;  }  public String getPassword() {  return password;  }  public void setPassword(String password) {  this.password = password;  }  @Id  private String userName;    @Column(name="password")  private String password;  } |