

Models :

Create a class Classes.java

```
package com.business.entity;

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

import java.util.List;

import javax.persistence.CascadeType;

@Entity
@Table(name="class_table")
public class Classes {

    @Id
    @GeneratedValue
    private int class_id;

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

    @OneToMany(cascade= CascadeType.ALL, mappedBy="classes")
    private List<Student> students;

    @OneToMany(cascade= CascadeType.ALL, mappedBy="classes")
    private List<Subject> subjects;

    public List<Student> getStudents() {
        return students;
    }

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

    public List<Subject> getSubjects() {
        return subjects;
    }

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

    public int getClass_id() {
        return class_id;
    }
}
```

```

        public void setClass_id(int class_id) {
            this.class_id = class_id;
        }

        public String getName() {
            return name;
        }

        public void setName(String name) {
            this.name = name;
        }
    }
}

```

Create a class Student.java

```

package com.business.entity;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
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
    private int student_id;

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

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

    @ManyToOne(cascade=CascadeType.ALL)
    @JoinColumn(name="class_id")
    private Classes classes;

    public int getStudent_id() {
        return student_id;
    }

    public void setStudent_id(int student_id) {

```

```

        this.student_id = student_id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getFname() {
        return fname;
    }

    public void setFname(String fname) {
        this.fname = fname;
    }

    public Classes getClasses() {
        return classes;
    }

    public void setClasses(Classes classes) {
        this.classes = classes;
    }
}

```

Create a class Subject.java

```

package com.business.entity;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.Table;

@Entity
@Table(name="subject_table")
public class Subject {

    @Id
    @GeneratedValue
    private int subject_id;

    @Column(name="subject_name")
    private String name;
}

```

```

@ManyToOne(cascade= CascadeType.ALL)
@JoinColumn(name="teacher_id")
private Teacher teacher;

@ManyToOne(cascade=CascadeType.ALL)
@JoinColumn(name="class_id")
private Classes classes;

public Classes getClasses() {
    return classes;
}

public void setClasses(Classes classes) {
    this.classes = classes;
}

public int getSubject_id() {
    return subject_id;
}

public void setSubject_id(int subject_id) {
    this.subject_id = subject_id;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public Teacher getTeacher() {
    return teacher;
}

public void setTeacher(Teacher teacher) {
    this.teacher = teacher;
}
}

```

Create a class Teacher.java

```

package com.business.entity;

import java.util.List;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.OneToOne;

```

```

import javax.persistence.Table;

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

    @Id
    @GeneratedValue
    private int teacher_id;

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

    @Column(name="last_name")
    private String lname;

    @OneToMany(cascade=CascadeType.ALL, mappedBy="teacher")
    private List<Subject> subjects;

    public int getTeacher_id() {
        return teacher_id;
    }

    public void setTeacher_id(int teacher_id) {
        this.teacher_id = teacher_id;
    }

    public String getName() {
        return name;
    }

    public void setName(String fname) {
        this.name = fname;
    }

    public String getLname() {
        return lname;
    }

    public void setLname(String lname) {
        this.lname = lname;
    }

    public List<Subject> getSubjects() {
        return subjects;
    }

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

}

```

Controllers :

Create a servlet AddClass.java

```
package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Classes;
import com.business.entity.Subject;
import com.business.util.HibernateUtil;

@WebServlet("/addClass")
public class AddClass extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AddClass() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("class");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();
```

```

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        //Step4; Create persistent object and add Teacher

        Classes classes = new Classes();
        classes.setName(name);

        session.save(classes);

        // Step5: Commit transaction and close session
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
        getServletContext().getRequestDispatcher("/viewClass.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create a servlet AddStudent.java

```

package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Student;
import com.business.util.HibernateUtil;
@WebServlet("/addStudent")
public class AddStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AddStudent() {
        super();
        // TODO Auto-generated constructor stub
    }
}

```

```

    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("fname");
        String lname = request.getParameter("lname");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        //Step4; Create persistent object and add Teacher

        Student student = new Student();
        student.setName(name);
        student.setFname(lname);

        session.save(student);

        // SStep5: Commit transaction and close sessoin
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewStudent.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create a servlet AddSubject.java

```

package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;

```



```

import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Subject;
import com.business.entity.Teacher;
import com.business.util.HibernateUtil;

@WebServlet("/addSubject")
public class AddSubject extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AddSubject() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("name");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        //Step4; Create persistent object and add Teacher

```

```

        Subject subject = new Subject();
        subject.setName(name);

        session.save(subject);

        // Step5: Commit transaction and close session
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewSubject.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create a servlet AddTeacher.java

```

package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Teacher;
import com.business.util.HibernateUtil;
@WebServlet("/addTeacher")
public class AddTeacher extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AddTeacher() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
    }
}

```

```

        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("name");
        String lname = request.getParameter("lname");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        //Step4; Create persistent object and add Teacher

        Teacher teacher = new Teacher();
        teacher.setName(name);
        teacher.setLname(lname);

        session.save(teacher);

        // STep5: Commit transaction and close sessoin
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewTeachers.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create a servlet AssignStudent.java

```

package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.RequestDispatcher;

```

```

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

import com.business.entity.Classes;
import com.business.entity.Student;
import com.business.util.HibernateUtil;
@WebServlet("/assignStudent")
public class AssignStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AssignStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("name");
        String[] nameList = name.split(" ");
        //System.out.println("NameList: " + nameList);
        String classes = request.getParameter("class");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        String hql_classes= "from Classes where name='" + classes + "'";
        List<Classes> clas = session.createQuery(hql_classes).list();

```

```

        String hql_student = "update Student s set s.classes=:n where
s.name=:sn and s.fname=:fn";

        Query<Student> query = session.createQuery(hql_student);
        query.setParameter("n", clas.get(0));
        query.setParameter("sn", nameList[0]);
        query.setParameter("fn", nameList[1]);

        query.executeUpdate();

        // SStep5: Commit transaction and close session
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewStudent.jsp");
        dispatcher.forward(request, response);
    }

}

```

Create a servlet AssignSubject.java

```

package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Classes;
import com.business.entity.Subject;
import com.business.entity.Teacher;
import com.business.util.HibernateUtil;
@WebServlet("/assignSubject")
public class AssignSubject extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AssignSubject() {

```

```

        super();
    }

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

        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String clas = request.getParameter("class");
        String subject = request.getParameter("subject");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        String hql_clas= "from Classes where name='" + clas + "'";
        List<Classes> classes = session.createQuery(hql_clas).list();

        String hql_subject = "update Subject s set s.classes=:n where
s.name=:sn";

        Query<Subject> query = session.createQuery(hql_subject);
        query.setParameter("n", classes.get(0));
        query.setParameter("sn", subject);

        query.executeUpdate();

        // SStep5: Commit transaction and close sessoin
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewClass.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create a servlet AssignTeacher.java

```
package com.business.servlet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

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

import com.business.entity.Classes;
import com.business.entity.Student;
import com.business.entity.Subject;
import com.business.entity.Teacher;
import com.business.util.HibernateUtil;

@WebServlet("/assignTeacher")
public class AssignTeacher extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public AssignTeacher() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at: ").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        // Step 1: Get details , user has entered
        String name = request.getParameter("name");
        String[] nameArray = name.split(" ");
```

```

        //System.out.println("NameList: " + nameList);
        String subject = request.getParameter("subject");

        // Step2: Create session
        SessionFactory sf = HibernateUtil.buildSessionFactory();
        Session session = sf.openSession();

        // Step 3: Begin Transaction
        Transaction tx = session.beginTransaction();

        String hql_teacher= "from Teacher where name like '%" +
nameArray[0] + "%'" + " and lname like '%" + nameArray[1] + "%'";
        // System.out.println("hql_teacher: "+hql_teacher);
        Query quer = session.createQuery(hql_teacher);
        //System.out.println("quer_str: "+quer.getQueryString());
        //System.out.println("quer: "+quer.getResultList());
        List<Teacher> teachers = quer.list();

        String hql_subject = "update Subject s set s.teacher=:n where
s.name=:sn";

        Query<Subject> query = session.createQuery(hql_subject);
        query.setParameter("n", teachers.get(0));
        query.setParameter("sn", subject);

        query.executeUpdate();

        // Step5: Commit transaction and close session
        tx.commit();
        session.close();

        RequestDispatcher dispatcher =
getServletContext().getRequestDispatcher("/viewTeachers.jsp");
        dispatcher.forward(request, response);
    }
}

```

Create aclass HibernateUtil.java for annotations

```

package com.business.util;

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

import com.business.entity.Classes;
import com.business.entity.Student;
import com.business.entity.Subject;
import com.business.entity.Teacher;

public class HibernateUtil {

    public static SessionFactory buildSessionFactory() {

```



```

        SessionFactory sessionFactory = new
Configuration().configure("hibernate.cfg.xml")
        .addAnnotatedClass(Teacher.class)
        .addAnnotatedClass(Subject.class)
        .addAnnotatedClass(Classes.class)
        .addAnnotatedClass(Student.class)
        .buildSessionFactory();

        return sessionFactory;
    }
}

```

Views & Display :

Create a html file addClasses.html

```

<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Add Class</h1>
    <form action="AddClass" method="post">
        <label for="class">Class:</label><br>
        <input type="text" id="class" name="class"><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>

```

Create a html file addStudent.html

```

<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Add Student</h1>
    <form action="AddStudent" method="post">
        <label for="fname">Name</label><br>
        <input type="text" id="fname" name="fname"><br>
        <label for="lname">Last Name</label><br>
        <input type="text" id="lname" name="lname"><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>

```

Create a html file addSubject.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Add Subject</h1>
    <form action="AddSubject" method="post">
        <label for="name">Subject Name:</label><br>
        <input type="text" id="name" name="name"><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Create a html file addTeacher.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Add Teacher</h1>
    <form action="AddTeacher" method="post">
        <label for="name">Name:</label><br>
        <input type="text" id="name" name="name"><br>
        <label for="lname">Last name</label><br>
        <input type="text" id="lname" name="lname"><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Create a JSP file index.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>Insert title here</title>
</head>
<body>

<h1>Welcome to administrative portal for the Learner Academy</h1>
<h3>What you want to do?</h3>

<table>
<tr>
    <th>Add Items</th>
    <th>View Items</th>
```

```

</tr>

<tr>

    <td><a href="addClasses.html">Add Class</a><br/></td>
    <td><a href="viewClass.jsp">View Classes</a><br/></td>
</tr>
<tr>
    <td><a href="addSubject.html">Add Subject</a><br/></td>
    <td><a href="viewSubject.jsp">View Subjects</a><br/></td>
</tr>
<tr>
    <td><a href="addTeacher.html">Add Teacher</a><br/></td>
    <td><a href="viewTeachers.jsp">View Teachers</a><br/></td>
</tr>
<tr>
    <td><a href="addStudent.html">Add Student</a><br/></td>
    <td><a href="viewStudent.jsp">View Students</a><br/></td>
</tr>
</table>
<br>

<a href="assignStudentToClass.jsp" class="card">Assign Student to
Class</a><br>
<a href="assignTeacherToSubject.jsp" class="card">Assign Teacher to
Class</a><br>
<a href="assignSubjectToClass.jsp" class="card">Assign Subject to
Class</a><br>

<a href="chooseClassReport.jsp">Choose Class for report</a>

</body>
</html>

```

Create a JSP file assignStudentToClass.jsp

```

<%@page import="com.business.entity.Student"%>
<%@page import="com.business.entity.Teacher"%>
<%@page import="com.business.entity.Subject"%>
<%@page import="com.business.entity.Classes"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>
<%String clas = request.getParameter("class");%>

```

```

<h1>Class Report for <%=clas%></h1>

<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Classes> classes = hibernateSession.createQuery("from Classes cl
where cl.name='" + clas + "'").list();
    Classes clasForReport = classes.get(0);
%>
<table>
<tr>
<th>Subject Name </th>
<th>Teacher Name </th>
</tr>
<%
    for (Subject subject: clasForReport.getSubjects()){
        out.print("<tr>");
        out.print("<td>" + subject.getName() + "</td>");
        out.print("<td>" + getTeacherName(subject) + "</td>");
        out.print("</tr>");
    }
%>
</table>
<%!
    public String getTeacherName(Subject subject){
        Teacher teacher = subject.getTeacher();
        String name;
        String lname;

        if (teacher != null){
            name=teacher.getName();
            lname = teacher.getLname();

            return name + " " + lname;
        }else{
            return "No Teacher assigned";
        }
    }
%>
<table>
<br>
<tr>
<th>Listed Students</th>
</tr>
<%
    for (Student student : clasForReport.getStudents()){
        out.print("<tr>");
        out.print("<td>" + student.getName() + " " + student.getFname() +
"</td>");
        out.print("</tr>");
    }
%>
</table>
</body>
</html>

```

Create a JSP file assignSubjectToClass.jsp

```
<%@page import="com.business.entity.Subject"%>
<%@page import="com.business.entity.Classes"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>

<h1>Assign a Subject to a Class</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Classes> classes = hibernateSession.createQuery("from
Classes").list();
    List<Subject> subjects = hibernateSession.createQuery("from
Subject").list();
%>
<form action="AssignSubject" method="post">
<table>
<tr>
<th>Class Name </th>
<th>Subject Name </th>
</tr>
<tr>
<td>
<select name="class">
<%
    for (Classes clas : classes){
        out.print("<option>" + clas.getName());
        out.print("</option>");
    }
%>
</select>
</td>
<td>
<select name="subject">
<%
    for (Subject subject : subjects){
        out.print("<option>" + subject.getName());
        out.print("</option>");
    }
%>
```

```

</select>
</td>
</tr>
</table>
<input type="submit" value="Submit">
</form>

```

```

</body>
</html>

```

Create a JSP file assignTeacherToSubject.jsp

```

<%@page import="com.business.entity.Subject"%>
<%@page import="com.business.entity.Teacher"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>

<h1>Assign a Teacher to a Subject</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Teacher> teachers = hibernateSession.createQuery("from
Teacher").list();
    List<Subject> subjects = hibernateSession.createQuery("from
Subject").list();
%>

<form action="AssignTeacher" method="post">
<table>
<tr>
<th>Student Name </th>
<th>Subject Name </th>
</tr>
<tr>
<td>
<select name="name">
<%
    for (Teacher teacher : teachers){
        out.print("<option>" + teacher.getName() + " " + teacher.getLname());
        out.print("</option>");
    }
%>
</select>

```

```

</td>

<td>
<select name="subject">
<%
    for (Subject subject : subjects){
        out.print("<option>" + subject.getName());
        out.print("</option>");
    }
%>
</select>
</td>
</tr>
</table>
<input type="submit" value="Submit">
</form>
</body>
</html>

```

Create a JSP file chooseClassReport.jsp

```

<%@page import="com.business.entity.Classes"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>

<h1>Choose a Class for Report Page</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Classes> classes = hibernateSession.createQuery("from
Classes").list();
%>

<form action="ClassReport.jsp" method="post">
<table>
<tr>
<th>Class Name </th>

</tr>

<tr>
<td>
<select name="class">

```

```

<%
    for (Classes clas : classes){
        out.print("<option>" + clas.getName());
        out.print("</option>");
    }
%>
</select>
</td>
</tr>
</table>
    <input type="submit" value="Submit">
</form>
</body>
</html>

```

Create a JSP file ClassReport.jsp

```

<%@page import="com.business.entity.Student"%>
<%@page import="com.business.entity.Teacher"%>
<%@page import="com.business.entity.Subject"%>
<%@page import="com.business.entity.Classes"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>
<%String clas = request.getParameter("class");%>
<h1>Class Report for <%=clas%></h1>

<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Classes> classes = hibernateSession.createQuery("from Classes cl
where cl.name='" + clas + "'").list();
    Classes clasForReport = classes.get(0);
%>
<table>
<tr>
<th>Subject Name </th>
<th>Teacher Name </th>
</tr>
<%
    for (Subject subject: clasForReport.getSubjects()){
        out.print("<tr>");
        out.print("<td>" + subject.getName() + "</td>");
        out.print("<td>" + getTeacherName(subject) + "</td>");
        out.print("</tr>");
    }
%>

```



```

    }
%>
</table>
<%!
    public String getTeacherName(Subject subject){
        Teacher teacher = subject.getTeacher();
        String name;
        String lname;

        if(teacher != null){
            name=teacher.getName();
            lname = teacher.getLname();

            return name + " " + lname;
        }else{
            return "No Teacher assigned";
        }
    }
%>
<table>
<br>
<tr>
<th>Listed Students</th>
</tr>
<%
    for(Student student : clasForReport.getStudents()){
        out.print("<tr>");
        out.print("<td>" + student.getName() + " " + student.getFname() +
"</td>");
        out.print("</tr>");
    }
%>
</table>
</body>
</html>

```

Create a JSP file viewClass.jsp

```

<%@page import="com.business.entity.Subject"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.entity.Classes"%>
<%@page import="java.util.List"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

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

<title>Insert title here</title>
</head>
<body>

```

```

<a href="index.jsp" class="card">Back to Main Menu</a><br>
<a href="addClasses.html" class="card">Add Class</a><br>
<a href="assignSubjectToClass.jsp" class="card">Assign Subject to
Class</a><br>

```

```

<h1>The Following Classes are listed</h1>

```

```

<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Classes> classes = hibernateSession.createQuery("from
Classes").list();
%>
<table>
<tr>
<th>Class Name </th>
<th>Subject Name</th>
</tr>
<%
    for(Classes clas : classes){
        out.print("<tr>");
        out.print("<td>" + clas.getName() + "</td>");

        StringBuffer buf = new StringBuffer();
        boolean first = true;
        for (Subject subject : clas.getSubjects()){
            if(first== true){
                buf.append("<td>" + subject.getName() +
"</td>");

                buf.append("</tr>");
                first = false;
            }else{
                buf.append("<tr><td></td>");
                buf.append("<td>" + subject.getName() +
"</td>");

                buf.append("</tr>");
            }
        }
        out.print(buf.toString());
    }
%>
</table>

</body>
</html>

```

Create a JSP file viewStudent.jsp

```

<%@page import="com.business.entity.Student"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>

```

```

<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>
<a href="addStudent.html">Add Student</a><br>
<a href="assignStudentToClass.jsp">Assign Student</a>

<h1>The Following Students are listed</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Student> students = hibernateSession.createQuery("from
Student").list();
%>
<table>
<tr>
<th>Name </th>
<th>Last Name </th>
<th>Assigned Class </th>
</tr>
<%
    for(Student student : students){
        out.print("<tr>");
        out.print("<td>" + student.getName() + "</td>");
        out.print("<td>" + student.getFname() + "</td>");
        out.print("<td>" + getClassName(student) + "</td>");
        out.print("</tr>");
    }
%>
</table>

<%!
    public String getClassName(Student student){
        if (student.getClasses() == null){
            return "no class assigned";
        }
        else{
            return student.getClasses().getName();
        }
    }
%>
</body>
</html>

```

Create a JSP file viewSubject.jsp

```
<%@page import="com.business.entity.Subject"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>
<a href="addSubject.html">Add Subject</a><br>

<h1>The Following Subjects are listed</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Subject> subjects = hibernateSession.createQuery("from
Subject").list();
%>
<table>
<tr>
<th>Subject Name </th>
</tr>
<%
    for(Subject subject : subjects){
        out.print("<tr>");
        out.print("<td>" + subject.getName() + "</td>");
        out.print("</tr>");
    }
%>
</table>

</body>
</html>
```

Create a JSP file viewTeacher.jsp

```
<%@page import="com.business.entity.Subject"%>
<%@page import="com.business.entity.Teacher"%>
<%@page import="java.util.List"%>
<%@page import="org.hibernate.Session"%>
<%@page import="com.business.util.HibernateUtil"%>
<%@page import="org.hibernate.SessionFactory"%>

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

```

<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<a href="index.jsp">Back to Main Menu</a><br>
<a href="addTeacher.html">Add Teacher</a><br>
<a href="assignTeacherToSubject.jsp">Assign Teacher to Subject</a><br>

<h1>The Following Teachers are listed</h1>
<%
    SessionFactory sf = HibernateUtil.buildSessionFactory();
    Session hibernateSession = sf.openSession();
    List<Teacher> teachers = hibernateSession.createQuery("from
Teacher").list();
%>
<table>
<tr>
<th>First Name </th>
<th>Last Name </th>
<th>Assigned Class</th>
</tr>
<%
    for(Teacher teacher : teachers){
        out.print("<tr>");
        out.print("<td>" + teacher.getName() + "</td>");
        out.print("<td>" + teacher.getLname() + "</td>");

        StringBuffer buf = new StringBuffer();
        boolean first = true;
        for (Subject subject : teacher.getSubjects()){
            if(first== true){
                buf.append("<td>" + subject.getName() +
"</td>");

                buf.append("</tr>");
                first = false;
            }else{
                buf.append("<tr><td></td><td></td>");
                buf.append("<td>" + subject.getName() +
"</td>");

                buf.append("</tr>");
            }
        }
        out.print(buf.toString());
    }
%>
</table>
</body>
</html>

```

Mapping with MySql :

Create a xml file hibernate.cfg.xml file

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
~ Hibernate, Relational Persistence for Idiomatic Java
~
~ License: GNU Lesser General Public License (LGPL), version 2.1 or later.
~ See the lgpl.txt file in the root directory or
http://www.gnu.org/licenses/lgpl-2.1.html.
-->
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
    <session-factory>

        <property name="format_sql">false</property>

        <property
name="connection.url">jdbc:mysql://localhost:3306/LearnerAcademy</property>
        <property
name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property name="connection.username">root</property>
        <property name="connection.password">Ponmalai11</property>

        <!-- JDBC connection pool (use the built-in) -->
        <property name="connection.pool_size">5</property>

        <!-- SQL dialect -->
        <property
name="dialect">org.hibernate.dialect.MySQL5Dialect</property>

        <!-- Disable the second-level cache -->
        <property
name="cache.provider_class">org.hibernate.cache.internal.NoCacheProvider</prop
erty>

        <!-- Echo all executed SQL to stdout -->
        <property name="show_sql">true</property>

        <!-- Drop and re-create the database schema on startup -->
        <property name="hbm2ddl.auto">create-drop</property>
        <property name="hbm2ddl.auto">update</property>
```

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

Pom.xml file for dependencies:

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>Phase-2</groupId>
  <artifactId>LearnerAcademy</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>

  <name>LearnerAcademy Maven Webapp</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>

    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>5.5.6.Final</version>
    </dependency>

    <!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>8.0.28</version>
    </dependency>

  </dependencies>

  <build>
```

```

    <finalName>LearnerAcademy</finalName>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven
defaults (may be moved to parent pom) -->
      <plugins>
        <plugin>
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- see http://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_war_packaging -->
        <plugin>
          <artifactId>maven-resources-plugin</artifactId>
          <version>3.0.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-compiler-plugin</artifactId>
          <version>3.8.0</version>
        </plugin>
        <plugin>
          <artifactId>maven-surefire-plugin</artifactId>
          <version>2.22.1</version>
        </plugin>
        <plugin>
          <artifactId>maven-war-plugin</artifactId>
          <version>3.2.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-install-plugin</artifactId>
          <version>2.5.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-deploy-plugin</artifactId>
          <version>2.8.2</version>
        </plugin>
      </plugins>
    </pluginManagement>
  </build>
</project>

```