

App.dao Pac

```
CourseDao.java
package app.dao;
import java.io.Serializable;
import java.util.List;
import org.hibernate.Session;
import org.hibernate.Transaction;
import app.entity.Course;
import app.hibernate.HibernateUtil;
public class CourseDao implements Serializable{
    Transaction transaction;
    public CourseDao() {}
    @SuppressWarnings("unchecked")
    public List<Course> fetchAllCourses() {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            return (List<Course>) session.createQuery("FROM Course").getResultList();
        } catch(Exception e) {
            System.out.println(e);
        }
        return null;
    }
    public Course getCourse(int course_id) {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            return session.get(Course.class, course_id);
        } catch(Exception e) {
            System.out.println(e);
        }
        return null;
    }
    public void deleteCourse(int course_id) {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            transaction = session.beginTransaction();
            Course course = session.get(Course.class, course_id);
            session.delete(course);
            transaction.commit();
        } catch(Exception e) {
            System.out.println(e);
        }
    }
    public void updateCourse(Course course) {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            transaction = session.beginTransaction();
            session.update(course);
            transaction.commit();
        } catch(Exception e) {
            System.out.println(e);
        }
    }
}
```

DepartmentDao.java

```
package app.dao;
import java.io.Serializable;
import java.util.List;
import org.hibernate.Session;
import org.hibernate.Transaction;
import app.entity.Department;
import app.hibernate.HibernateUtil;
public class DepartmentDao implements Serializable{
    Transaction transaction;
    public DepartmentDao() {
    }
    public List<Department> fetchAllDepartmentList() {
```

```

        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            return session.createQuery("FROM Department").getResultList();
        } catch( Exception e) {
            System.out.println(e);
        }
        return null;
    }
}

```

EnrollmentsDao.java

```

package app.dao;
import java.util.List;
import org.hibernate.Session;
import app.entity.Enrollment;
import app.hibernate.HibernateUtil;
public class EnrollmentsDao {
    public EnrollmentsDao() {}
    public List<Enrollment> getEnrollments() {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            List<Enrollment> enrollmentList = (List<Enrollment>)
session.createQuery("FROM Enrollment").getResultList();

            for(Enrollment e: enrollmentList) {
                System.out.println(e.getStudent().getFirst_name());
            }
            return enrollmentList;
        } catch(Exception e) {
            System.out.println(e);
        }
        return null;
    }
}

```

StudentDao.java

```

package app.dao;
import java.io.Serializable;
import java.util.List;
import org.hibernate.Session;
import org.hibernate.Transaction;
import app.entity.Students;
import app.hibernate.HibernateUtil;
public class StudentDao implements Serializable{
    Transaction transaction;
    public StudentDao() {
    }

    public boolean updateStudent(Students student) {
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            transaction = session.beginTransaction();
            session.update(student);
            transaction.commit();
            return true;
        } catch(Exception e) {
            System.out.println(e);
        }
        return false;
    }
    @SuppressWarnings("unchecked")
    public List<Students> fetchAllStudents() {
        try(Session session = HibernateUtil.getSessionFactory().openSession()) {
            return (List<Students>) session.createQuery("FROM Students").getResultList();
        } catch(Exception e) {
            System.out.println(e);
        }
        return null;
    }
}

```

```

public boolean saveStudent(Students student) {
try(Session session = HibernateUtil.getSessionFactory().openSession()) {
transaction = session.beginTransaction();
session.save(student);
transaction.commit();
return true;
} catch(Exception e) {
System.out.println(e);
}return false;
}

public boolean deleteStudent(int student_id) {
try (Session session = HibernateUtil.getSessionFactory().openSession()) {
transaction = session.beginTransaction();
Students student = session.get(Students.class, student_id);
session.delete(student);
transaction.commit();
return true;
} catch(Exception e) {
System.out.println(e);
} return false;
}

public Students getStudent(int student_id) {
try(Session session = HibernateUtil.getSessionFactory().openSession()) {
return session.get(Students.class, student_id);
} catch(Exception e) {
System.out.println(e);
} return null;
}
}

```

UserDao.java

```

package app.dao;
import java.io.Serializable;
import java.nio.charset.spi.CharsetProvider;
import org.hibernate.Session;
import app.classes.MD5;
import app.entity.User;
import app.hibernate.HibernateUtil;
public class UserDao implements Serializable{
public boolean authenticate(User user) {
try(Session session = HibernateUtil.getSessionFactory().openSession()) {
User dbUser = (User) session.createQuery("FROM User WHERE username
= :username").setParameter("username", user.getUsername()).getSingleResult();
return (dbUser.getPassword().equals(MD5.getMd5(user.getPassword())));
} catch(Exception e) {
System.out.println(e);
}
return false;
}
}

```

App.entity

User.java

```

package app.entity;
import java.io.Serializable;
import java.time.LocalDate;
import javax.persistence.*;
@Entity
@Table(name = "users")

```

```

public class User implements Serializable {
    public User () {}
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "user_id")
    private int user_id;
    @Column(name = "username", length = 50, unique = true)
    private String username;
    @Column(name = "password", length = 255)
    private String password;
    @Column(name = "role", length = 20)
    private String role;
    @Column(name = "created_at")
    private LocalDate created_at;
    public int getUser_id() {
        return user_id;
    }
    public void setUser_id(int user_id) {
        this.user_id = user_id;
    }
    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;
    }
    public String getRole() {
        return role;
    }
    public void setRole(String role) {
        this.role = role;
    }
    public LocalDate getCreated_at() {
        return created_at;
    }
    public void setCreated_at(LocalDate created_at) {
        this.created_at = created_at;
    }
}

```

WEB-INF

View

Login.jsp

```

<%@page import="org.hibernate.internal.build.AllowSysOut"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Login - Student Management System</title>

<link
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
  rel="stylesheet">
  <%
String message = "";
if(request.getParameter("auth") != null) {
    message = "Invalid username or password";
}
%>
</head>
<body class="d-flex justify-content-center align-items-center vh-100">
<div class="card p-4" style="width: 30rem;">
    <h3 class="text-center mb-4">Staff Login</h3>
    <p style="color:red; font-weight: 400;"><%=message %></p>
    <form action="checklogin" method="POST">
        <div class="mb-3">
            <label for="username" class="form-label">Username</label> <input
                type="text" class="form-control" id="username" name="username"
                required>
        </div>
        <div class="mb-3">
            <label for="password" class="form-label">Password</label> <input
                type="password" class="form-control" id="password" name="password"
                required>
        </div>
        <button type="submit" class="btn btn-primary w-100">Login</button>
    </form>
</div>
<script
  src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js
/bootstrap.bundle.min.js"></script>
</body>
</html>

```

Dashboard.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Dashboard - Student Management System</title>
<!-- Bootstrap CSS -->
<link
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
  rel="stylesheet">
</head>
<body>
    <div class="container py-5">
        <h1 class="text-center mb-5">Welcome to the Dashboard</h1>
        <div class="row">

```

```

<div class="col-md-4">
<div class="card">
<div class="card-body">
<h5 class="card-title">Manage Students</h5>
<p class="card-text">View and manage student records.</p>
<a href="students" class="btn btn-primary">Go to Students</a>
</div>
</div>
</div>
<div class="col-md-4">
<div class="card">
<div class="card-body">
<h5 class="card-title">Manage Courses</h5>
<p class="card-text">View and manage courses.</p>
<a href="courses" class="btn btn-primary">Go to Courses</a>
</div> </div>
</div>
<div class="col-md-4">
<div class="card">
<div class="card-body">
<h5 class="card-title">Manage Enrollments</h5>
<p class="card-text">Enroll students in courses.</p>
<a href="enrollments" class="btn btn-primary">Go to
Enrollments</a>
</div>
</div> </div> </div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist
/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

Courses.jsp

```

<%@page import="app.entity.Course"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Courses - Student Management System</title>
<!-- Bootstrap CSS -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<%
List<Course> courseList = (List<Course>) request.getAttribute("courseList");
%>

```

```

<div class="container py-5">
    <h1 class="mb-4">Courses List</h1>
    <button class="btn btn-warning" data-bs-toggle="modal"
    data-bs-target="#studentModal">Add Course</button>

    <a href="dashboard" class="btn btn-secondary m-1">Close</a>
    <table class="table table-bordered">
        <thead>
            <tr>
                <th>Course ID</th>
                <th>Course Name</th>
                <th>Course Code</th>
                <th>Credits</th>
                <th>Department</th>
                <!-- New column for department -->
                <th>Actions</th>
            </tr>
        </thead>
        <tbody>
            <%for(Course c: courseList) {
            %>
            <tr>
                <td><%=c.getCourse_id() %></td>
                <td><%=c.getCourse_name() %></td>
                <td><%=c.getCourse_code() %></td>
                <td><%=c.getCredits() %></td>
                <td><%=c.getDepartment().getDepartment_name() %></td>
                <!-- Displaying department name -->
                <td><a href="editCourse?cid=<%=c.getCourse_id() %>" class="btn btn-warning btn-sm">Edit</a>
                <a href="deleteCourse?cid=<%=c.getCourse_id() %>" class="btn btn-danger btn-sm">Delete</a>
            </td>
            </tr>
            <% } %>
        </tbody>
    </table>
</div>

<!-- Bootstrap JS -->
<script
    src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1
/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

editCourse.jsp

```

<%@page import="app.entity.Course"%>
<%@page import="app.entity.Department"%>
<%@page import="java.util.List"%>
<%@page import="app.entity.Students"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>

```

```

<meta charset="UTF-8">
<title>Edit Student</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<%
List<Department> departmentList = (List<Department>) request.getAttribute("departmentList");
Course course = (Course) request.getAttribute("course");
%>
<div class="modal-content" style="margin: 50px; width: 70%;">
    <!-- Modal Header -->
    <div class="modal-header">
        <h5 class="modal-title" id="studentModalLabel">Enter Course Details</h5>
    </div>

    <!-- Modal Body with Form -->
    <div class="modal-body">
        <form id="studentForm" method="POST" action="update-course">

            <div class="mb-3">
                <label for="courseid" class="form-label">Course ID</label>
                <input
                    disabled
                    type="text" class="form-control"
                    value="<%=course.getCourse_id() %>" name="course_id">
            </div>
            <div class="mb-3">
                <label for="lastname" class="form-label">Course Name</label> <input
                    type="text" class="form-control" name="coursename"
                    value="<%=course.getCourse_name() %>" required>
            </div>
            <div class="mb-3">
                <label for="email" class="form-label">Course Code</label> <input
                    type="text" class="form-control" name="coursecode"
                    value="<%=course.getCourse_code() %>" required>
            </div> </div>
            <div class="mb-3">
                <label for="gender" class="form-label">Department</label> <select
                    class="form-select" name="department" required>

                    <%for(Department d: departmentList) { %>
                        <option value="<%=d.getDepartment_id() %>" <%=d.getDepartment_id()
== course.getDepartment_id() ? "selected" : ""%>><%=d.getDepartment_name() %></option>

                    <%}%>
                </select>
            </div>
            <div class="mb-3">
                <label for="phone" class="form-label">Credits</label> <input

```

```

        type="text" class="form-control" name="credits"
        value="<%>course.getCredits()%<%>" required>
    </div></form>
</div>
<div class="modal-footer">
    <a type="button" href="courses" class="btn btn-secondary m-1" data-bs-dismiss="modal">Close</a>
    <button type="submit" class="btn btn-primary" form="studentForm">Update Student</button>
</div>
</div>
</body>
</html>

```

Students.jsp

```

<%@page import="java.util.ArrayList"%>
<%@page import="app.entity.Students"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Students - Student Management System</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<%
List<Students> studentList = (List<Students>)request.getAttribute("studentList");
if(studentList == null) studentList = new ArrayList();
%>
<body>
<div class="container py-5">
    <h1 class="mb-4">Student List</h1>
    <button class="btn btn-warning" data-bs-toggle="modal" data-bs-target="#studentModal">Add Student</button>

    <a href="dashboard" class="btn btn-secondary m-1">Close</a>
    <br><br>
    <table class="table table-bordered">
        <thead>
            <tr>
                <th>Name</th>
                <th>Email</th>
                <th>Enrollment Date</th>
                <th>Actions</th>
            </tr>
        </thead>
        <tbody>
            <%
            for(Students s: studentList) {
%>

```

```

<tr>
<td><%=s.getFirst_name() + " " + s.getLast_name() %></td>
<td><%=s.getEmail() %></td>
<td><%=s.getEnrollement_date() %></td>
<td><a href="editStudent?sid=<%=s.getStudent_id() %>" class="btn btn-warning btn-sm">Edit</a>
<a href="deleteStudent?sid=<%=s.getStudent_id() %>" class="btn btn-danger btn-sm">Delete</a>
</td>
</tr>
<%
}
%>
</tbody>
</table>
<div class="modal fade" id="studentModal" tabindex="-1"
aria-labelledby="studentModalLabel" aria-hidden="true">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<h5 class="modal-title" id="studentModalLabel">Enter Student
Details</h5>
<button type="button" class="btn-close" data-bs-dismiss="modal"
aria-label="Close"></button>
</div>
<div class="modal-body">
<form id="studentForm" method="POST" action="save-student">
<div class="mb-3">
<label for="firstname" class="form-label">First Name</label> <input
type="text" class="form-control" id="firstname"
name="firstname" required>
</div>
<div class="mb-3">
<label for="lastname" class="form-label">Last Name</label> <input
type="text" class="form-control" id="lastname" name="lastname"
required>
</div>
<div class="mb-3">
<label for="email" class="form-label">Email</label> <input
type="email" class="form-control" id="email" name="email"
required>
</div>
<div class="mb-3">
<label for="dob" class="form-label">Date of
Birth</label> <input type="date" class="form-control" id="dob"
name="dob" required>
</div>
<div class="mb-3">
<label for="gender" class="form-label">Gender</label> <select
class="form-select" id="gender" name="gender" required>
<option value="" selected disabled>Select Gender</option>
<option value="M">Male</option>
<option value="F">Female</option>
</select>
</div>
<div class="mb-3">
<label for="phone" class="form-label">Phone</label> <input
type="text" class="form-control" id="phone" name="phone"

```

```

required>
</div>
<div class="mb-3">
<label for="address" class="form-label">Address</label>
<textarea class="form-control" id="address" name="address" rows="3" required></textarea>
</div>
<div class="mb-3">
<label for="enrollmentdate" class="form-label">Enrollment Date</label> <input type="date" class="form-control" id="enrollmentdate" name="enrollmentdate" required>
</div></form>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
<button type="submit" class="btn btn-primary" form="studentForm">Save Student</button>
</div> </div>
</div></div>
</div>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

editStudent.jsp

```

<%@page import="app.entity.Students"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Edit Student</title>
<link
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
  rel="stylesheet">
</head>
<body>
<%Students student = (Students) request.getAttribute("student"); %>
<div class="modal-content" style="margin: 50px; width: 70%;>
  <div class="modal-header">
    <h5 class="modal-title" id="studentModalLabel">Enter Student Details</h5>
  </div>
  <div class="modal-body">
    <form id="studentForm" method="POST" action="update-student">
      <div class="mb-3">
        <label for="firstname" class="form-label">First Name</label>
        <input value="<%student.getStudent_id() %>" name="sid" type="hidden"/>
        <input
          type="text" class="form-control" id="firstname"
          value="<%student.getFirst_name() %>">
        <input
          name="firstname" required>
      </div>
    </form>
  </div>
</div>

```

```

</div>
<div class="mb-3">
<label for="lastname" class="form-label">Last Name</label> <input
type="text" class="form-control" id="lastname" name="lastname"
value="<%student.getLast_name()%>"
required>
</div>
<div class="mb-3">
<label for="email" class="form-label">Email</label> <input
type="email" class="form-control" id="email" name="email"
value="<%student.getEmail()%>"
required>
</div>
<div class="mb-3">
<label for="dob" class="form-label">Date of
Birth</label> <input type="date" class="form-control" id="dob"
value="<%student.getDate_of_birth()%>"
name="dob" required>
</div>
<div class="mb-3">
<label for="gender" class="form-label">Gender</label> <select
class="form-select" id="gender" name="gender" required>
<option value="M" <%=(student.getGender() == 'M')? "Selected": "" %>>Male</option>
<option value="F" <%=(student.getGender() == 'F')? "Selected": "" %>>Female</option>
</select>
</div>
<div class="mb-3">
<label for="phone" class="form-label">Phone</label> <input
type="text" class="form-control" id="phone" name="phone"
value="<%student.getPhone()%>"
required>
</div>
<div class="mb-3">
<label for="address" class="form-label">Address</label>
<textarea class="form-control" id="address" name="address"
rows="3" required><%student.getAddress()%</textarea>
</div>
<div class="mb-3">
<label for="enrollmentdate" class="form-label">Enrollment
Date</label> <input type="date" class="form-control"
value="<%student.getEnrollement_date()%>"
id="enrollmentdate" name="enrollmentdate" required>
</div> </form>
</div>
<div class="modal-footer">
<a type="button" href="students" class="btn btn-secondary m-1"
data-bs-dismiss="modal">Close</a>
<button type="submit" class="btn btn-primary" form="studentForm">Update
Student</button>
</div> /div>
</body>
</html>

```

3.IF port already reserve use this

netstat -ano | findstr :8080