**Project Detail**

Project Name: Learner’s Academy

**Project Details:** This project is for the design and developing a backend administrative portal for the Learner’s Academy using Java EE technologies

**Objective**: The Objective of this project is to apply servlet, jsp and JDBC concepts.

**Developer Detail**

Name: Harish Ragaventhar C

**Technologies and tools Used:**

* Servlet: to do the business logic and works a controller for the project.
* JSP: to handle the presentation view.
* SQL: to create and manage the database.
* JDBC: to make operations on the database for the project.
* Eclipse: to write and run the code.
* Tomcat: to run and deploy servlet application.

**Git Hub Repository:**

https://github.com/harishragaventhar/phase2-learners-login.git

**Algorithm:**

1. Create a new database in your SQL server to store the student and teacher data.
2. Create two tables: "Students" and "Teachers", with appropriate fields such as name,classroom,subject and travel,
3. A database connection object in your servlet using the JDBC driver.
4. Use the database connection object to query the database and retrieve data.
5. Add a form to the JSP page with fields for the email and password.
6. Create a Java servlet that will handle the login request.
7. Retrieve the email and password from the form data.
8. Query the database to check if the email and password match.
9. create a session and redirect the user to the appropriate page (student or teacher dashboard).
10. If the authentication fails, redirect the user back to the login page with an error message a JSP page for the student dashboard.
11. Display the student's information retrieved from the database.
12. Add links to view the student's courses and grade
13. Create a JSP page for the teacher dashboard.
14. Add links to view the teacher's courses and students.

**Flow Chart:**



**Sprint Plan:**

**Sprint 1:**

1. Create database and Tables.
2. Connect Database to the project
3. Create models classes.
4. Create a database utility class to retrieve data.
5. Create login page.

**Sprint 2:**

1. Create JSP Files for all pages of the project.
2. Create a servlet to get requests and responses to the JSP files.

**Concept used in this programme:**

1. **Object-Oriented:** used to create and model objects for users and their credentials.
2. **Databases**: used to store and retrieve data.
3. **Data Sources:** used to define a set of properties required to identify and access the database.
4. **Collections:** used some collections such array list to store collection of data.
5. **Exception Handling**: used to catch problems that arises in the code especially in I/O blocks.
6. **Cookies:** to store log-in data on the client browser. Searching and Sorting

**How to run the program:**

1. Import the “database\database.sql” file to your database administration tool.
2. Go to “\src\main\webapp\WEB-INF\META-INF context.xml” file and open it.
3. Edit the database’ properties such as email, password and driverClassName to be suit to your database administration tool.
4. Now run program on a server.
5. To login you must enter admin for both username and password.

**Code:**

package database;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import entities.ClassRoom;

import entities.Student;

import entities.Subject;

import entities.Teacher;

public class ClassDatabase {

public boolean insertClass(String name, String day, String time, int subjectId, int teacherId, int studentId) throws SQLException

{

String sql = " insert into class (name, day, time, subject\_id, teacher\_id, student\_id)" + " values(?,?,?,?,?,?)";

System.out.println(sql);

// 1. DB connection

Connection conn = DBConnection.dbConn();

//2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, name);

stat.setString(2, day);

stat.setString(3, time);

stat.setInt(4, subjectId);

stat.setInt(5, teacherId);

stat.setInt(6, studentId);

try {

stat.execute();

}catch(Exception e)

{

System.out.println("error "+e.getMessage());

return false;

}

return true;

}

public List<ClassRoom> getAllClasses() throws SQLException {

List<ClassRoom> classes = new ArrayList<ClassRoom>();

String sql = "select \* from class";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

Statement stat = conn.createStatement();

// 3. execute the query

ResultSet rs = stat.executeQuery(sql);

while (rs.next()) {

ClassRoom classRoom = new ClassRoom();

Subject subject = new Subject();

SubjectDatabase sDb = new SubjectDatabase();

Teacher teacher = new Teacher();

TeacherDatabase tDb = new TeacherDatabase();

Student student = new Student();

StudentDatabase studDb = new StudentDatabase();

classRoom.setId(rs.getInt(1));

classRoom.setName(rs.getString(2));

classRoom.setDay(rs.getString(3));

classRoom.setTime(rs.getString(4));

subject = sDb.getSubjectById(rs.getInt(5));

classRoom.setSubject(subject);

student = studDb.getStudentById(rs.getInt(6));

classRoom.setStudent(student);

teacher = tDb.getTeacherById(rs.getInt(7));

classRoom.setTeacher(teacher);

classes.add(classRoom);

}

return classes;

}

}

package database;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBConnection {

public static Connection dbConn()

{

Connection conn = null;

try {

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

System.out.println("Driver loaded");

conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/",

"root", "Simplilearn");

System.out.println("DB conected");

} catch (ClassNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return conn;

}

}

package database;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import entities.Student;

import entities.Subject;

public class StudentDatabase {

public boolean insertStudent(Student student) throws SQLException {

String sql = " insert into student (name)" + " values(?)";

System.out.println(sql);

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, student.getName());

try {

stat.execute();

} catch (Exception e) {

System.out.println("error " + e.getMessage());

return false;

}

return true;

}

public List<Student> getAllStudents() throws SQLException {

List<Student> students = new ArrayList<Student>();

String sql = "select \* from student";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

Statement stat = conn.createStatement();

// 3. execute the query

ResultSet rs = stat.executeQuery(sql);

while (rs.next()) {

Student student = new Student();

student.setId(rs.getInt(1));

student.setName(rs.getString(2));

students.add(student);

}

return students;

}

public int getStudentIdByName(String name) throws SQLException {

String sql = "select \* from student where name=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, name);

// 3. execute the query

ResultSet rs = stat.executeQuery();

// int subjectId = 0;

// if (rs.next()) {

// subjectId = rs.getInt(rs.getInt(1));

// }

Student student = new Student();

while (rs.next()) {

student.setId(rs.getInt(1));

student.setName(rs.getString(2));

}

return student.getId();

}

public Student getStudentById(int id) throws SQLException {

String sql = "select \* from student where id=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

//2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setInt(1, id);

// 3. execute the query

Student s = null;

ResultSet rs = stat.executeQuery();

while(rs.next()) {

s= new Student();

s.setId(rs.getInt(1));

s.setName(rs.getString(2));

}

return s;

}

}package database;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import entities.Student;

import entities.Subject;

public class StudentDatabase {

public boolean insertStudent(Student student) throws SQLException {

String sql = " insert into student (name)" + " values(?)";

System.out.println(sql);

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, student.getName());

try {

stat.execute();

} catch (Exception e) {

System.out.println("error " + e.getMessage());

return false;

}

return true;

}

public List<Student> getAllStudents() throws SQLException {

List<Student> students = new ArrayList<Student>();

String sql = "select \* from student";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

Statement stat = conn.createStatement();

// 3. execute the query

ResultSet rs = stat.executeQuery(sql);

while (rs.next()) {

Student student = new Student();

student.setId(rs.getInt(1));

student.setName(rs.getString(2));

students.add(student);

}

return students;

}

public int getStudentIdByName(String name) throws SQLException {

String sql = "select \* from student where name=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, name);

// 3. execute the query

ResultSet rs = stat.executeQuery();

// int subjectId = 0;

// if (rs.next()) {

// subjectId = rs.getInt(rs.getInt(1));

// }

Student student = new Student();

while (rs.next()) {

student.setId(rs.getInt(1));

student.setName(rs.getString(2));

}

return student.getId();

}

public Student getStudentById(int id) throws SQLException {

String sql = "select \* from student where id=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

//2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setInt(1, id);

// 3. execute the query

Student s = null;

ResultSet rs = stat.executeQuery();

while(rs.next()) {

s= new Student();

s.setId(rs.getInt(1));

s.setName(rs.getString(2));

}

return s;

}

}

package database;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import entities.Subject;

import entities.Teacher;

public class TeacherDatabase {

public boolean insertTeacher(Teacher teacher) throws SQLException {

String sql = " insert into teacher (name, email)" + " values(?,?)";

System.out.println(sql);

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, teacher.getName());

stat.setString(2, teacher.getEmail());

try {

stat.execute();

} catch (Exception e) {

System.out.println("error " + e.getMessage());

return false;

}

return true;

}

public List<Teacher> getAllTeachers() throws SQLException {

List<Teacher> teachers = new ArrayList<Teacher>();

String sql = "select \* from teacher";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

Statement stat = conn.createStatement();

// 3. execute the query

ResultSet rs = stat.executeQuery(sql);

while (rs.next()) {

Teacher teacher = new Teacher();

teacher.setId(rs.getInt(1));

teacher.setName(rs.getString(2));

teacher.setEmail(rs.getString(3));

teachers.add(teacher);

}

return teachers;

}

public int getTeacherIdByName(String name) throws SQLException {

String sql = "select \* from teacher where name=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

// 2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, name);

// 3. execute the query

ResultSet rs = stat.executeQuery();

Teacher teacher = new Teacher();

// int teacherId = 0;

// if (rs.next()) {

// teacherId = rs.getInt(rs.getInt(2));

// }

while (rs.next()) {

teacher.setId(rs.getInt(1));

teacher.setName(rs.getString(2));

teacher.setEmail(rs.getString(3));

}

return teacher.getId();

}

public Teacher getTeacherById(int id) throws SQLException {

String sql = "select \* from teacher where id=?";

// 1. DB connection

Connection conn = DBConnection.dbConn();

//2. create the statememt

PreparedStatement stat = conn.prepareStatement(sql);

stat.setInt(1, id);

// 3. execute the query

Teacher t = null;

ResultSet rs = stat.executeQuery();

while(rs.next()) {

t= new Teacher();

t.setId(rs.getInt(1));

t.setName(rs.getString(2));

t.setEmail(rs.getString(3));

}

return t;

}

}

package database;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import entities.User;

public class UserDatabase {

public boolean validateUser(String email, String password) throws SQLException

{

boolean isValid = false;

String sql = "select password from user where email=?";

Connection conn = DBConnection.dbConn();

PreparedStatement stat = conn.prepareStatement(sql);

stat.setString(1, email);

ResultSet rs = stat.executeQuery();

if(rs.next()) {

String pass = rs.getString(1);

if(pass.equals(password))

isValid = true;

}

return isValid;

}

public List<User> getAllUsers() throws SQLException

{

List<User> users = new ArrayList<User>();

String sql = "select \* from user";

// 1. DB connection

Connection conn = DBConnection.dbConn();

//2. create the statememt

Statement stat = conn.createStatement();

// 3. execute the query

ResultSet rs = stat.executeQuery(sql);

while(rs.next()) {

User user = new User();

user.setEmail(rs.getString(1));

user.setPhone(rs.getString(2));

user.setPassword(rs.getString(3));

user.setCity(rs.getString(4));

users.add(user);

}

return users;

}

}

package servlets;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

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 database.ClassDatabase;

import database.StudentDatabase;

import database.SubjectDatabase;

import database.TeacherDatabase;

import entities.ClassRoom;

import entities.Student;

import entities.Subject;

import entities.Teacher;

/\*\*

\* Servlet implementation class RegisterServlet

\*/

@WebServlet("/classes")

public class ClassesServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ClassesServlet() {

super();

System.out.println("Classes Servlet");

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

SubjectDatabase sDB = new SubjectDatabase();

TeacherDatabase tDB = new TeacherDatabase();

StudentDatabase studDB = new StudentDatabase();

try {

List<Subject> subjects = sDB.getAllSubjects();

List<Teacher> teachers = tDB.getAllTeachers();

List<Student> students = studDB.getAllStudents();

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

request.setAttribute("subjects", subjects);

request.setAttribute("teachers", teachers);

request.setAttribute("students", students);

dispatcher.forward(request, response);

} catch (SQLException e) {

e.printStackTrace();

}

}

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String name = request.getParameter("name");

String day = request.getParameter("day");

String time = request.getParameter("time");

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

String teacherName = request.getParameter("teacher");

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

SubjectDatabase sdb = new SubjectDatabase();

TeacherDatabase tdb = new TeacherDatabase();

StudentDatabase studdb = new StudentDatabase();

System.out.println(name);

System.out.println(day);

System.out.println(time);

System.out.println(subjectName);

System.out.println(teacherName);

System.out.println(studentName);

try {

System.out.println(sdb.getSubjectIdByName(subjectName));

System.out.println(tdb.getTeacherIdByName(teacherName));

System.out.println(studdb.getStudentIdByName(studentName));

} catch (SQLException e2) {

// TODO Auto-generated catch block

e2.printStackTrace();

}

if (name == null || name.isEmpty()) {

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

request.setAttribute("error", "Name cannot be null");

dispatcher.forward(request, response);

return;

}

if (day == null || day.isEmpty()) {

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

request.setAttribute("error", "Day cannot be null");

dispatcher.forward(request, response);

return;

}

if (time == null || time.isEmpty()) {

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

request.setAttribute("error", "Time cannot be null");

dispatcher.forward(request, response);

return;

}

ClassDatabase db = new ClassDatabase();

ClassRoom classRoom = new ClassRoom();

classRoom.setName(name);

classRoom.setDay(day);

classRoom.setTime(time);

String nameToInsert = name;

String dayToInsert = day;

String timeToInsert = time;

int subjectIdToInsert = 0;

int teacherIdToInsert = 0;

int studentToInsert = 0;

try {

subjectIdToInsert = sdb.getSubjectIdByName(subjectName);

teacherIdToInsert = tdb.getTeacherIdByName(teacherName);

studentToInsert = studdb.getStudentIdByName(studentName);

System.out.println(subjectIdToInsert);

System.out.println(teacherIdToInsert);

} catch (SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

try {

if (db.insertClass(nameToInsert, dayToInsert, timeToInsert, subjectIdToInsert, teacherIdToInsert, studentToInsert)) {

// take me to which page? login page

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding class successfully</p>");

dispatcher.include(request, response);

} else {

// response.sendRedirect("register.html");

RequestDispatcher dispatcher = request.getRequestDispatcher("classes.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding class unsuccessful</p>");

dispatcher.include(request, response);

}

} catch (SQLException e) {

// keep me on registration page

response.sendRedirect("classes.jsp");

}

}

}

package servlets;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

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 javax.servlet.http.HttpSession;

import database.UserDatabase;

/\*\*

\* Servlet implementation class LoginServlet

\*/

@WebServlet("/login")

public class LoginServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public LoginServlet() {

System.out.println("Login Servlet");

}

/\*\*

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

\*/

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

response.sendRedirect("login.jsp");

}

/\*\*

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

\*/

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

response.setHeader("Cache-Control", "no-cache,no-store,must-revalidate");

response.setHeader("Pragma", "no-cache");//http1.0

response.setHeader("Pragma", "0");//proxies

String email = request.getParameter("email");

String password = request.getParameter("password");

PrintWriter out = response.getWriter();

response.setContentType("text/html");

// try {

// if(email!=null && email.equals("admin")

// && password!=null && password.equals("admin")) {

// request.setAttribute("email", email);

// // url rewriting

// HttpSession session = request.getSession();

// session.setAttribute("id", email);

// response.sendRedirect("dashboard.jsp");

// }

// else {

// response.sendRedirect("login.jsp?error=Invalid Credentials");

// }

// } catch (Exception e) {

// // TODO Auto-generated catch block

// e.printStackTrace();

// response.sendRedirect("login.jsp?error=something went wrong");

// }

if(email!=null && email.equals("admin")

&& password!=null && password.equals("admin") )

{

HttpSession session = request.getSession();

session.setAttribute("id", email);

response.sendRedirect("classes");

return;

}

UserDatabase db = new UserDatabase();

try {

if(db.validateUser(email, password)) {

request.setAttribute("email", email);

// url rewriting

HttpSession session = request.getSession();

session.setAttribute("id", email);

response.sendRedirect("dashboard.jsp");

}

else {

response.sendRedirect("login.jsp?error=Invalid Credentials");

}

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

response.sendRedirect("login.jsp?error=something went wrong");

}

}

}

package servlets;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

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 database.StudentDatabase;

import entities.Student;

/\*\*

\* Servlet implementation class StudentsServlets

\*/

@WebServlet("/students")

public class StudentsServlets extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public StudentsServlets() {

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

response.sendRedirect("students.jsp");

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// TODO Auto-generated method stub

String name = request.getParameter("name");

System.out.println(name);

if (name == null || name.isEmpty()) {

RequestDispatcher dispatcher = request.getRequestDispatcher("students.jsp");

request.setAttribute("error", "Name cannot be null");

dispatcher.forward(request, response);

return;

}

StudentDatabase db = new StudentDatabase();

Student student = new Student();

student.setName(name);

try {

if (db.insertStudent(student)) {

// take me to which page? login page

RequestDispatcher dispatcher = request.getRequestDispatcher("students.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding student successfully</p>");

dispatcher.include(request, response);

} else {

// response.sendRedirect("register.html");

RequestDispatcher dispatcher = request.getRequestDispatcher("students.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding student unsuccessful</p>");

dispatcher.include(request, response);

}

} catch (SQLException e) {

// keep me on registration page

response.sendRedirect("students.jsp");

}

}

}

package servlets;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

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 database.SubjectDatabase;

import entities.Subject;

/\*\*

\* Servlet implementation class RegisterServlet

\*/

@WebServlet("/subjects")

public class SubjectsServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public SubjectsServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

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

\*/

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

response.sendRedirect("subjects.jsp");

}

/\*\*

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

\*/

protected void doPost(HttpServletRequest request,

HttpServletResponse response) throws ServletException, IOException {

String name = request.getParameter("name");

System.out.println(name);

if(name == null || name.isEmpty())

{

RequestDispatcher dispatcher = request.getRequestDispatcher("subjects.jsp");

request.setAttribute("error", "Name cannot be null");

dispatcher.forward(request, response);

return;

}

SubjectDatabase db = new SubjectDatabase();

Subject subject = new Subject();

subject.setName(name);

try {

if(db.insertSubject(subject))

{

// take me to which page? login page

RequestDispatcher dispatcher = request.getRequestDispatcher("subjects.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding subject successfully</p>");

dispatcher.include(request, response);

}

else {

//response.sendRedirect("register.html");

RequestDispatcher dispatcher = request.getRequestDispatcher("subjects.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding subject unsuccessful</p>");

dispatcher.include(request, response);

}

} catch (SQLException e) {

// keep me on registration page

response.sendRedirect("subjects.jsp");

}

}

}

package servlets;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.SQLException;

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 database.TeacherDatabase;

import entities.Teacher;

/\*\*

\* Servlet implementation class RegisterServlet

\*/

@WebServlet("/teachers")

public class TeachersServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public TeachersServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

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

\*/

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

response.sendRedirect("teachers.jsp");

}

/\*\*

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

\*/

protected void doPost(HttpServletRequest request,

HttpServletResponse response) throws ServletException, IOException {

String name = request.getParameter("name");

String email = request.getParameter("email");

System.out.println(name);

System.out.println(email);

if(name == null || name.isEmpty())

{

RequestDispatcher dispatcher = request.getRequestDispatcher("teachers.jsp");

request.setAttribute("error", "Name cannot be null");

dispatcher.forward(request, response);

return;

}

if(email == null || email.isEmpty())

{

RequestDispatcher dispatcher = request.getRequestDispatcher("teachers.jsp");

request.setAttribute("error", "Email cannot be null");

dispatcher.forward(request, response);

return;

}

TeacherDatabase db = new TeacherDatabase();

Teacher teacher = new Teacher();

teacher.setName(name);

teacher.setEmail(email);

try {

if(db.insertTeacher(teacher))

{

// take me to which page? login page

RequestDispatcher dispatcher = request.getRequestDispatcher("teachers.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding teacher successfully</p>");

dispatcher.include(request, response);

}

else {

//response.sendRedirect("register.html");

RequestDispatcher dispatcher = request.getRequestDispatcher("teachers.jsp");

PrintWriter out = response.getWriter();

out.println("<p>Adding teacher unsuccessful</p>");

dispatcher.include(request, response);

}

} catch (SQLException e) {

// keep me on registration page

response.sendRedirect("teachers.jsp");

}

}

}

package servlets;

import java.io.IOException;

import java.sql.SQLException;

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 database.ClassDatabase;

import database.TeacherDatabase;

import entities.Teacher;

/\*\*

\* Servlet implementation class ViewClassesServlet

\*/

@WebServlet("/viewClasses")

public class ViewClassesServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ViewClassesServlet() {

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

ClassDatabase cDb = new ClassDatabase();

try {

List<entities.ClassRoom> classes = cDb.getAllClasses();

RequestDispatcher dispatcher = request.getRequestDispatcher("viewClasses.jsp");

request.setAttribute("classes", classes);

dispatcher.forward(request, response);

} catch (SQLException e) {

e.printStackTrace();

}

}

/\*\*

\* @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);

}

}

package servlets;

import java.io.IOException;

import java.sql.SQLException;

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 database.StudentDatabase;

import database.SubjectDatabase;

import entities.Student;

import entities.Subject;

/\*\*

\* Servlet implementation class ViewStudentsServlet

\*/

@WebServlet("/viewStudents")

public class ViewStudentsServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ViewStudentsServlet() {

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

StudentDatabase studDB = new StudentDatabase();

try {

List<Student> students = studDB.getAllStudents();

RequestDispatcher dispatcher = request.getRequestDispatcher("viewStudents.jsp");

request.setAttribute("students", students);

dispatcher.forward(request, response);

} catch (SQLException e) {

e.printStackTrace();

}

}

/\*\*

\* @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);

}

}

package servlets;

import java.io.IOException;

import java.sql.SQLException;

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 database.SubjectDatabase;

import entities.Subject;

/\*\*

\* Servlet implementation class ViewSubjectsServlet

\*/

@WebServlet("/viewSubjects")

public class ViewSubjectsServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ViewSubjectsServlet() {

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

SubjectDatabase sDB = new SubjectDatabase();

try {

List<Subject> subjects = sDB.getAllSubjects();

RequestDispatcher dispatcher = request.getRequestDispatcher("viewSubjects.jsp");

request.setAttribute("subjects", subjects);

dispatcher.forward(request, response);

} catch (SQLException e) {

e.printStackTrace();

}

}

/\*\*

\* @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);

}

}

package servlets;

import java.io.IOException;

import java.sql.SQLException;

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 database.SubjectDatabase;

import database.TeacherDatabase;

import entities.Subject;

import entities.Teacher;

/\*\*

\* Servlet implementation class ViewTeachersServlet

\*/

@WebServlet("/viewTeachers")

public class ViewTeachersServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public ViewTeachersServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

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

\*/

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

TeacherDatabase tDB = new TeacherDatabase();

try {

List<Teacher> teachers = tDB.getAllTeachers();

RequestDispatcher dispatcher = request.getRequestDispatcher("viewTeachers.jsp");

request.setAttribute("teachers", teachers);

dispatcher.forward(request, response);

} catch (SQLException e) {

e.printStackTrace();

}

}

/\*\*

\* @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);

}

}

<%@page import="entities.Subject"%>

<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

<style>

.error {

color: red;

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

</ul>

</div>

</div>

</nav>

<div class="container">

<h2 align="center">Please add a new class by entering its details</h2>

</div>

<div class="container">

<%

String error = (String) request.getAttribute("error");

String name = (String) request.getAttribute("name");

String day = (String) request.getAttribute("day");

String time = (String) request.getAttribute("time");

String subjectName = (String) request.getAttribute("subject");

String teacherName = (String) request.getAttribute("teacher");

if (name == null)

name = "";

if (day == null)

day = "";

if (time == null)

time = "";

if (error != null) {

%>

<div class="error"><%=error%></div>

<%

}

%>

<form action="classes" method="post">

<div class="row">

<div class="col-lg-6 col-lg-offset-3">

<div class="form-group">

<label for="name">Name:\*</label> <input type="text"

class="form-control" id="className" placeholder="Enter name"

name="name" value=<%=name%>>

</div>

<br>

<div class="form-group">

<label for="name">Day:\*</label> <input type="text"

class="form-control" id="classDay" placeholder="Enter day"

name="day" value=<%=day%>>

</div>

<br>

<div class="form-group">

<label for="name">Time:\*</label> <input type="text"

class="form-control" id="classTime" placeholder="Enter time"

name="time" value=<%=time%>>

</div>

<br>

<div class="drop-down">

<label for="name">Subject:</label><br> <select name="subject"

class="drop-down" id="subject" name="subjectName">

<c:forEach items="${requestScope.subjects}" var="subject">

<option>${subject.name}</option>

<font></font>

</c:forEach>

</select>

</div>

<br>

<div class="drop-down">

<label for="name">Teacher:</label><br> <select name="teacher"

class="drop-down" id="teacher" name="teacherName">

<c:forEach items="${requestScope.teachers}" var="teacher">

<option>${teacher.name}</option>

<font></font>

</c:forEach>

</select>

</div>

<br>

<div class="drop-down">

<label for="name">Student:</label><br> <select name="student"

class="drop-down" id="student" name="studentName">

<c:forEach items="${requestScope.students}" var="student">

<option>${student.name}</option>

<font></font>

</c:forEach>

</select>

</div>

<br>

<div align="center">

<input type="submit" class="btn btn-primary" value="Save" />

</div>

</div>

</div>

</form>

</div>

<!-- <div class="container">

<table class="table">

<tr>

<th>Id</th>

<th>Name</th>

</tr>

<c:forEach var="subject" items="${requestScope.subjects}">

<tr>

<td>${ subject.id}</td>

<td>${ subject.name}</td>

</tr>

</c:forEach>

</table>

</div> -->

}

</body>

</html>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

</ul>

</div>

</div>

</nav>

<h1>Logged in successfully!</h1>

</body>

</html>

<%-- JSP Page Directive --%>

<%@page import="java.time.LocalDateTime"%>

<%@page import="java.util.Date"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

<style>

p {

color: white;

background-color: black;

padding: 10px;

}

</style>

</head>

<body>

<!-- Java inside HTML

-->

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<!-- <li class="nav-item"><a class="nav-link active"

aria-current="page" href="index.html">Home</a></li> -->

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<h1>Learner's Academy</h1>

<%

// scriplets => inside a jspService()

out.println("<br> <p>" + "Welcome ADMIN! Today is: " + new Date() + "</p>");

%>

<%-- JSP Expressions

do not put ; at the end. it can converts everything strings

--%>

</body>

</html>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

<style>

.error{

color:red;

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<div class="container">

<div>

<div class="wrapper fadeInDown">

<div id="formContent">

<div class="fadeIn first">

<h2 class='sign'>Sign In</h2>

</div>

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

<input type="text" id="login" class="fadeIn second" name="email"

placeholder="Email" value="admin" /> <input type="password"

value="admin" id="password" class="fadeIn third" name="password"

placeholder="Password" /> <input type="submit"

class="fadeIn fourth" value="Log In" />

</form>

<%

String error = request.getParameter("error");

if(error != null){

%>

<div class="error"><%= error %></div>

<%} %>

</div>

</div>

</div>

</div>

</body>

</html>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<%

session.removeAttribute("id");

session.invalidate();

response.sendRedirect("login.jsp");

%>

</body>

</html>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

<style>

.error {

color: red;

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<div class="container">

<h2 align="center">Please add a new student by entering its name</h2>

</div>

<div class="container">

<%

String error = (String) request.getAttribute("error");

String name = (String) request.getAttribute("name");

if (name == null)

name = "";

if (error != null) {

%>

<div class="error"><%=error%></div>

<%

}

%>

<form action="students" method="post">

<div class="row">

<div class="col-lg-6 col-lg-offset-3">

<div class="form-group">

<label for="name">Student Name:\*</label> <input type="text"

class="form-control" id="studentName" placeholder="Enter Name"

name="name" value=<%=name%>>

</div>

<div align="center">

<input type="submit" class="btn btn-primary" value="Save" />

</div>

</div>

</div>

</form>

</div>

</body>

</html>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

<style>

.error {

color: red;

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<div class="container">

<h2 align="center">Please add a new teacher by entering its

information</h2>

</div>

<div class="container">

<%

String error = (String) request.getAttribute("error");

String name = (String) request.getAttribute("name");

String email = (String) request.getAttribute("email");

if (name == null)

name = "";

if (email == null)

email = "";

if (error != null) {

%>

<div class="error"><%=error%></div>

<%

}

%>

<form action="teachers" method="post">

<div class="row">

<div class="col-lg-6 col-lg-offset-3">

<div class="form-group">

<label for="name">Teacher's Name:\*</label> <input type="text"

class="form-control" id="teacherName" placeholder="Enter Name"

name="name" value=<%=name%>>

</div>

<div class="form-group">

<label for="name">Teacher's Email:\*</label> <input type="text"

class="form-control" id="teacherEmail" placeholder="Enter Email"

name="email" value=<%=email%>>

</div>

<div align="center">

<input type="submit" class="btn btn-primary" value="Save" />

</div>

</div>

</div>

</form>

</div>

</body>

</html>

<%@page import="entities.Student"%>

<%@page import="entities.Subject"%>

<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<%

List<Student> students = (List<Student>) request.getAttribute("students");

%>

<div class="container">

<table class="table">

<tr>

<th>Id</th>

<th>Name</th>

</tr>

<c:forEach var="student" items="${students}">

<tr>

<td>${ student.id}</td>

<td>${ student.name}</td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

<%@page import="entities.Subject"%>

<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<%

List<Subject> subjects = (List<Subject>) request.getAttribute("subjects");

%>

<div class="container">

<%

String subjectName = (String) request.getAttribute("subject");

%>

<table class="table">

<tr>

<th>Id</th>

<th>Name</th>

</tr>

<c:forEach var="subject" items="${subjects}">

<tr>

<td>${ subject.id}</td>

<td>${ subject.name}</td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

<%@page import="entities.Teacher"%>

<%@page import="entities.Subject"%>

<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

rel="stylesheet"

integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"

crossorigin="anonymous">

<script

src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"

crossorigin="anonymous"></script>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<a class="navbar-brand" href="#">LA</a>

<button class="navbar-toggler" type="button"

data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"

aria-controls="navbarSupportedContent" aria-expanded="false"

aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<%

String id = (String) session.getAttribute("id");

if (id == null) {

%>

<li class="nav-item"><a class="nav-link" href="login.jsp">Login</a>

</li>

<%

}

if (id != null) {

%>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="subjects.jsp">Subjects</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="teachers.jsp">Teachers</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="students.jsp">Students</a></li>

<li class="nav-item"><a class="nav-link active"

aria-current="page" href="classes">Classes</a></li>

<li class="nav-item"><a class="nav-link" href="viewSubjects">View

Subjects</a></li>

<li class="nav-item"><a class="nav-link" href="viewTeachers">View

Teachers</a></li>

<li class="nav-item"><a class="nav-link" href="viewStudents">View

Students</a></li>

<li class="nav-item"><a class="nav-link" href="viewClasses">View

Classes</a></li>

<li class="nav-item"><a class="nav-link" href="logout.jsp">Logout</a>

</li>

<%

}

%>

</ul>

</div>

</div>

</nav>

<%

List<Teacher> teachers = (List<Teacher>) request.getAttribute("teachers");

%>

<div class="container">

<%

String teacherName = (String) request.getAttribute("teacher");

%>

<table class="table">

<tr>

<th>Id</th>

<th>Name</th>

<th>Email</th>

</tr>

<c:forEach var="teacher" items="${teachers}">

<tr>

<td>${ teacher.id}</td>

<td>${ teacher.name}</td>

<td>${ teacher.email}</td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>