package com.learnersacademy.admin;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import javax.sql.DataSource;

import com.learnersacademy.models.Student;

import com.learnersacademy.models.Subject;

import com.learnersacademy.models.Teacher;

import com.learnersacademy.models.Class;

public class DbRetrieve {

private DataSource dataSource;

public DbRetrieve(DataSource dataSource) {

this.dataSource = dataSource;

}

public List<Student> getStudents() {

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

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

// get a connection

myConn = dataSource.getConnection();

// create sql stmt

String sql = "SELECT \* FROM students";

myStmt = myConn.createStatement();

// execute query

myRs = myStmt.executeQuery(sql);

// process result

while (myRs.next()) {

// retrieve data from result set row

int id = myRs.getInt("id");

String firstName = myRs.getString("fname");

String lastName = myRs.getString("lname");

int age = myRs.getInt("age");

int aclass = myRs.getInt("class");

// create new student object

Student tempStudent = new Student(id, firstName, lastName, age, aclass);

// add it to the list of students

students.add(tempStudent);

}

} catch (Exception e) {

// TODO: handle exception

} finally {

// close JDBC objects

close(myConn, myStmt, myRs);

}

return students;

}

public List<Teacher> getTeachers() {

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

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM teachers";

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

String firstName = myRs.getString("fname");

String lastName = myRs.getString("lname");

int age = myRs.getInt("age");

Teacher temp = new Teacher(id, firstName, lastName, age);

teachers.add(temp);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return teachers;

}

public List<Subject> getSubjects() {

List<Subject> subjects = new ArrayList<>();

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM subjects";

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

String name = myRs.getString("name");

String shortcut = myRs.getString("shortcut");

Subject temp = new Subject(id, name,shortcut);

subjects.add(temp);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return subjects;

}

public List<Class> getClasses() {

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

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM classes";

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

int section = myRs.getInt("section");

int subject = myRs.getInt("subject");

int teacher = myRs.getInt("teacher");

String time = myRs.getString("time");

Teacher tempTeacher = loadTeacher(teacher);

Subject tempSubject = loadSubject(subject);

String teacher\_name = tempTeacher.getFname() + " " + tempTeacher.getLname();

Class temp = new Class(id, section, teacher\_name, tempSubject.getName(), time);

classes.add(temp);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return classes;

}

public Teacher loadTeacher(int teacherId) {

Teacher theTeacher = null;

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM teachers WHERE id = " + teacherId;

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

String fname = myRs.getString("fname");

String lname = myRs.getString("lname");

int age = myRs.getInt("age");

theTeacher = new Teacher(id, fname, lname, age);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return theTeacher;

}

public Subject loadSubject(int subjectId) {

Subject theSubject = null;

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM subjects WHERE id = " + subjectId;

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

String name = myRs.getString("name");

String shortcut = myRs.getString("shortcut");

theSubject = new Subject(id, name,shortcut);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return theSubject;

}

public Class loadClass(int classId) {

Class theClass = null;

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM clasess WHERE id = " + classId;

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

int section = myRs.getInt("section");

int subject = myRs.getInt("subject");

int teacher = myRs.getInt("teacher");

String time = myRs.getString("time");

Teacher tempTeacher = loadTeacher(teacher);

Subject tempSubject = loadSubject(subject);

String teacher\_name = tempTeacher.getFname() + " " + tempTeacher.getLname();

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return theClass;

}

public List<Student> loadClassStudents(int classId) {

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

Connection myConn = null;

Statement myStmt = null;

ResultSet myRs = null;

try {

myConn = dataSource.getConnection();

String sql = "SELECT \* FROM students WHERE class = " + classId;

myStmt = myConn.createStatement();

myRs = myStmt.executeQuery(sql);

while (myRs.next()) {

int id = myRs.getInt("id");

String firstName = myRs.getString("fname");

String lastName = myRs.getString("lname");

int age = myRs.getInt("age");

int aclass = myRs.getInt("class");

Student tempStudent = new Student(id, firstName, lastName, age, aclass);

students.add(tempStudent);

}

} catch (Exception e) {

} finally {

close(myConn, myStmt, myRs);

}

return students;

}

private void close(Connection myConn, Statement myStmt, ResultSet myRs) {

try {

if (myRs != null) {

myRs.close();

}

if (myStmt != null) {

myStmt.close();

}

if (myConn != null) {

myConn.close();

}

} catch (Exception e) {

e.printStackTrace();

}

}

}