University of Makati
J.P. Rizal Extension, West Rembo
Makati City

STUDENT RECORD MANAGEMENT SYSTEM

Computer Programming 2

Presented by:
Magsino, Gino Ben

Presented to:
Prof. Christian Michael Mansueto
Subject Adviser

January 2020

I. OBJECTIVES

a. Create a program with a user-friendly interface.

b. Develop a program that stores student’s record which can be accessed or modified anytime.

c. Effectively apply the lessons learned in Computer Programming 1.

II. PROJECT DESCRIPTION

Student Records Management System is a program designed to store, administer and manage student information. This program can add student records (ID, Name and Grade) as well as removing or modifying them. The current program can only hold up to five students records but this limitation can be adjusted by the developer. The developer used Java programming language, Java Development Kit (JDK) Version 13.0.1 and Eclipse Java as its Integrated Development Environment (IDE). The program uses MySQL Version 8.0.18 as its database and is only connected on the device’s localhost. The program was made as a requirement in Computer Programming 2.

III. CONCEPTS/TOPICS APPLIED

1. While Loop

- Used to iterate on student records to show them on table. It was also used on the main method to make a loop after every function is executed unless the exit button is pressed.

2. If/else Statement

- To perform actions based on different conditions on the program. It is the most used topic and is mainly applied to show whether the Student ID that was entered exists on the database.

3. Switch Statement

- A control statement that allows execution which corresponds to the given value. It was only used on main method to execute multiple methods quickly.

4. JTable

- To organize data which has both rows and columns and display it in tabular form.

5. JScrollPane

- Used to scroll through different data from JTable.

6. **JOptionPane**

   - To display information or get inputs from user through dialog boxes.

7. **User Defined Function**

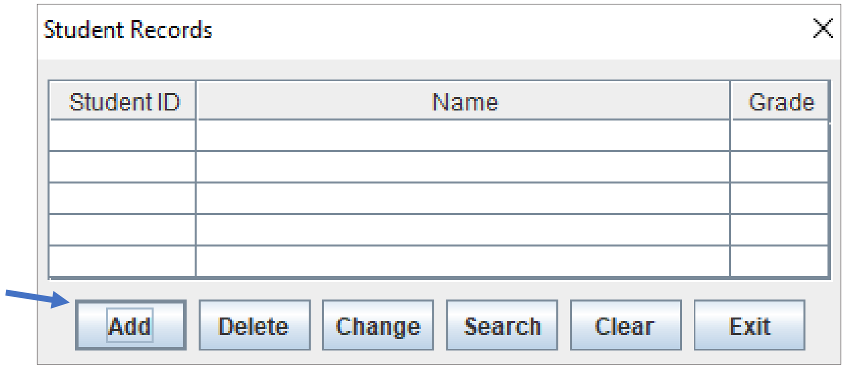
   - Used in creating different types of function on the program that helps shorten the code.

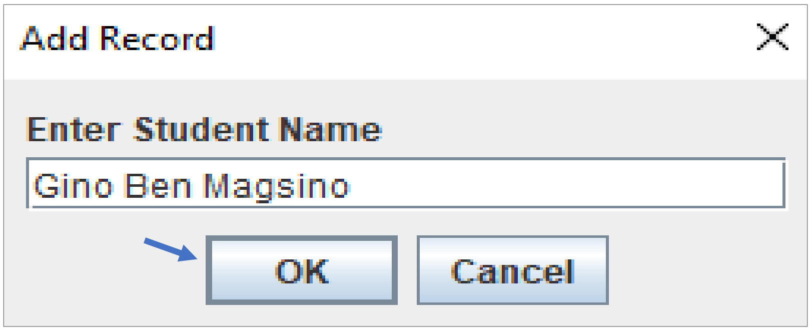
8. **MySQL**

   - A database management system that was used by the developed java program.

IV. SAMPLE TEST DATA

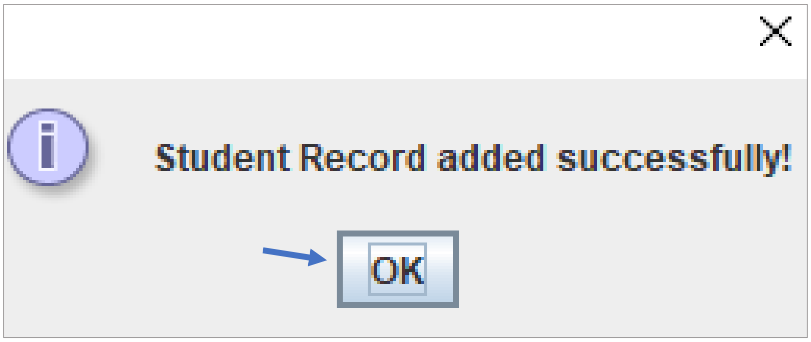
1. ADDING A STUDENT RECORD





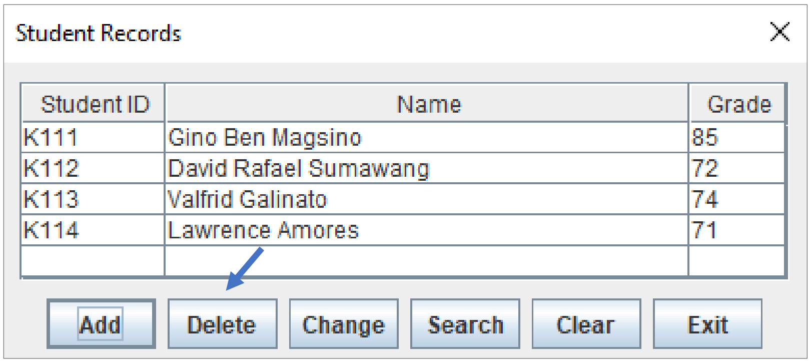
Enter Student Name

Gino Ben Magsino



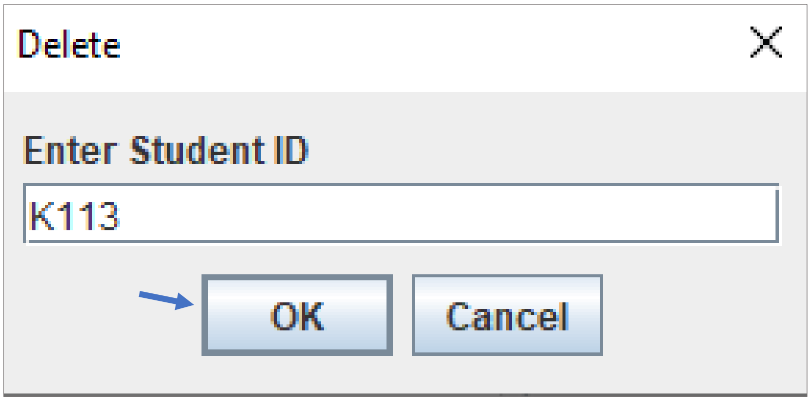
Student Record added successfully!

2. REMOVING A STUDENT'S RECORD



Student Records

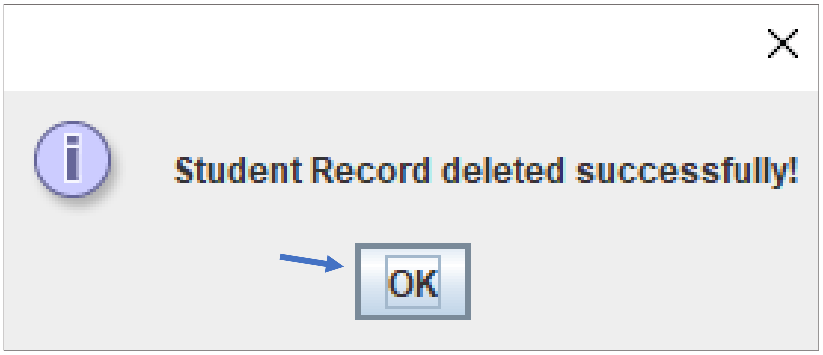
Student ID	Name	Grade
K111	Gino Ben Magsino	85
K112	David Rafael Sumawang	72
K113	Valfrid Galinato	74
K114	Lawrence Amores	71



Delete

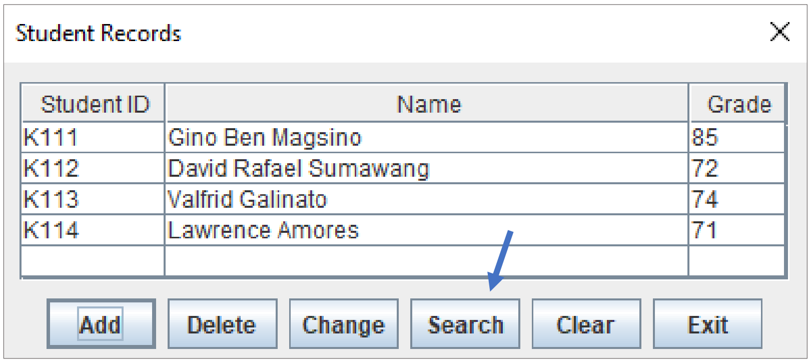
Enter Student ID

K113

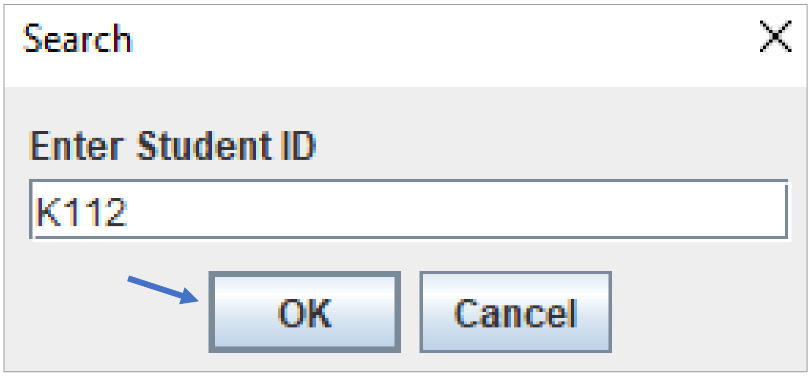


Student Record deleted successfully!

3. SEARCHING FOR STUDENT'S RECORD

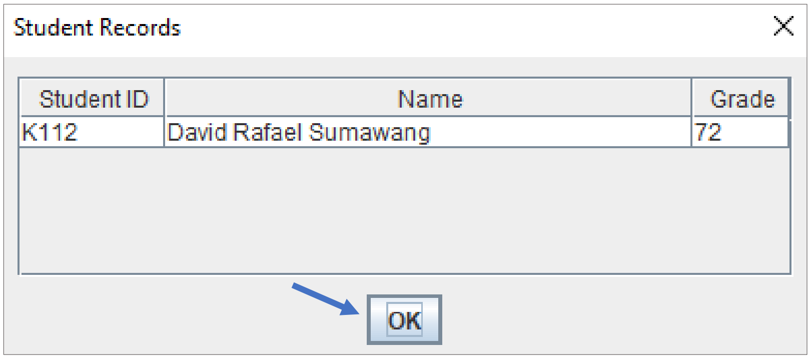


Student ID	Name	Grade
K111	Gino Ben Magsino	85
K112	David Rafael Sumawang	72
K113	Valfrid Galinato	74
K114	Lawrence Amores	71



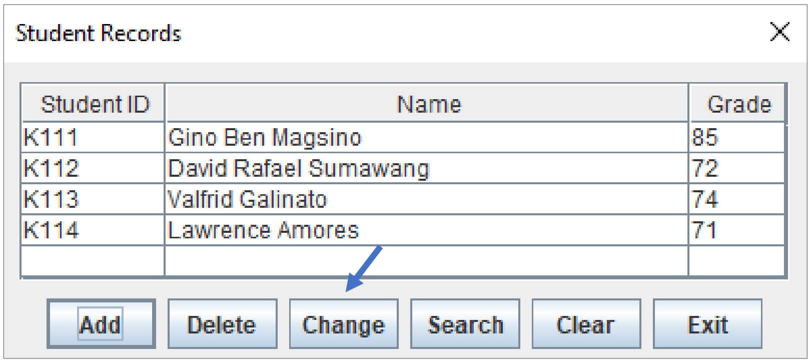
Enter Student ID

K112

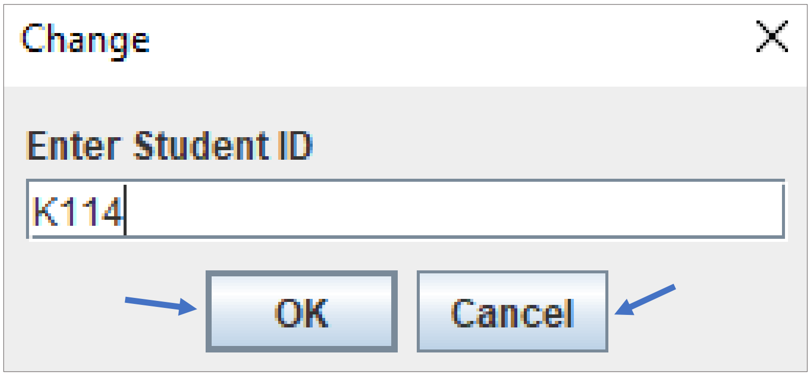


Student ID	Name	Grade
K112	David Rafael Sumawang	72

4. CHANGING A STUDENT'S GRADE

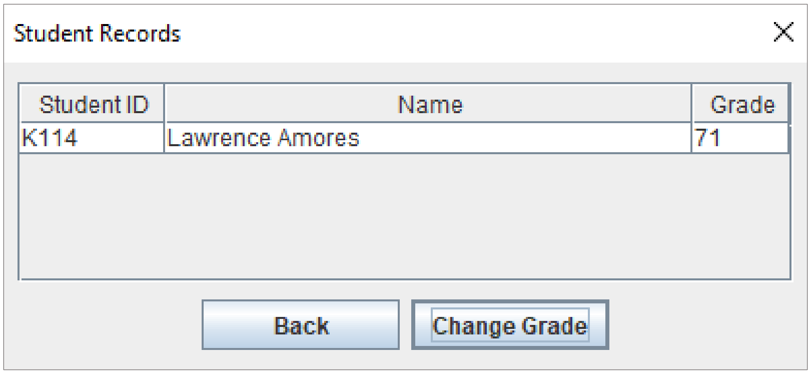

Student Records

Student ID	Name	Grade
K111	Gino Ben Magsino	85
K112	David Rafael Sumawang	72
K113	Valfrid Galinato	74
K114	Lawrence Amores	71

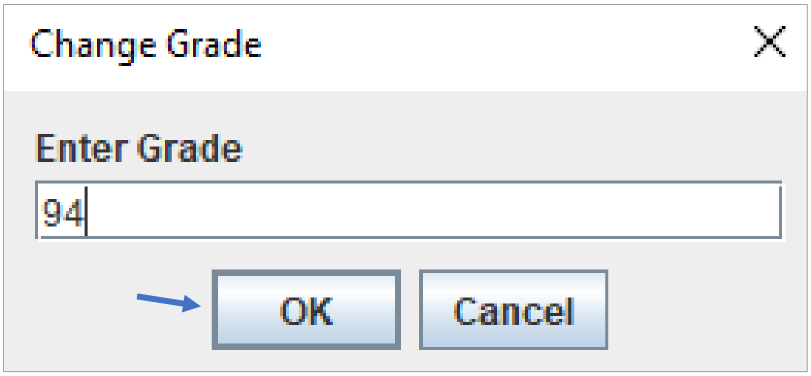

Change

Enter Student ID

K114

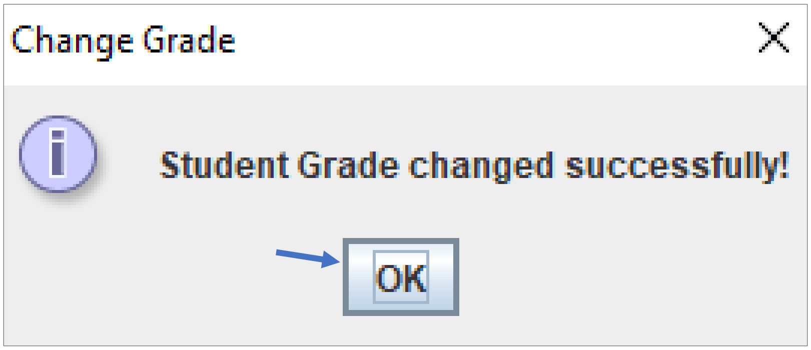

Student Records

Student ID	Name	Grade
K114	Lawrence Amores	71


Change Grade

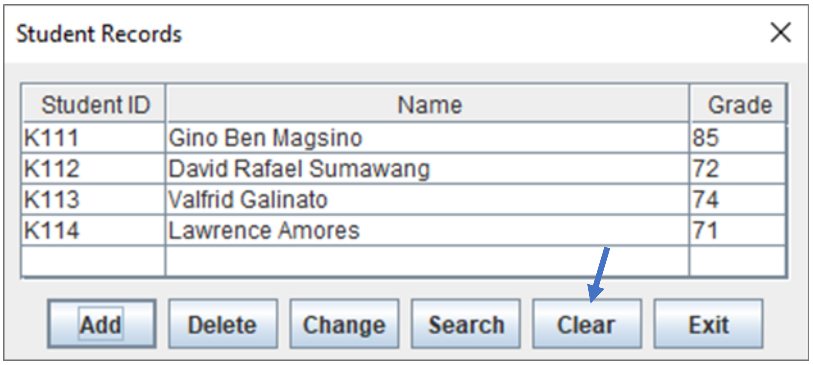
Enter Grade

94

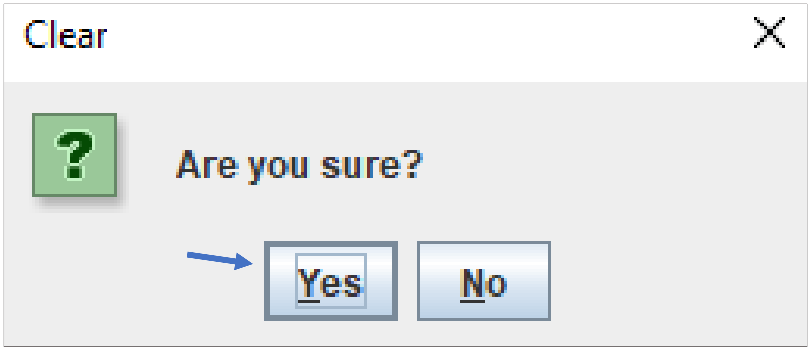

Change Grade

Student Grade changed successfully!

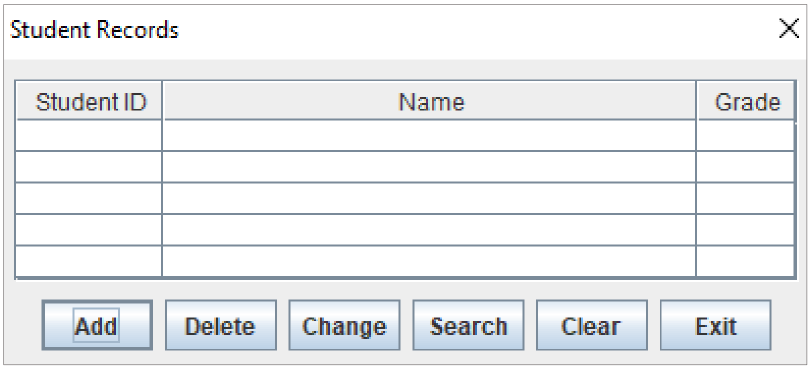
5. Clearing All Student's Record


Student Records

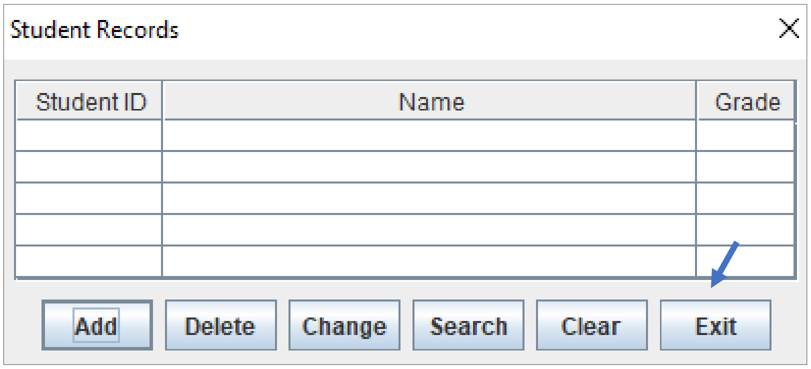
Student ID	Name	Grade
K111	Gino Ben Magsino	85
K112	David Rafael Sumawang	72
K113	Valfrid Galinato	74
K114	Lawrence Amores	71

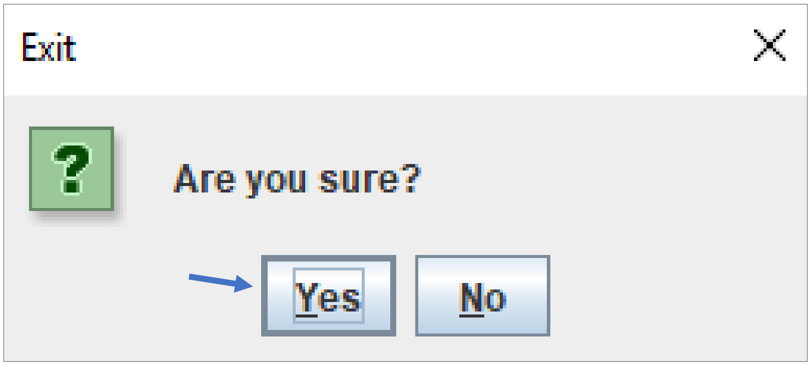


Are you sure?



6. Exiting the Program

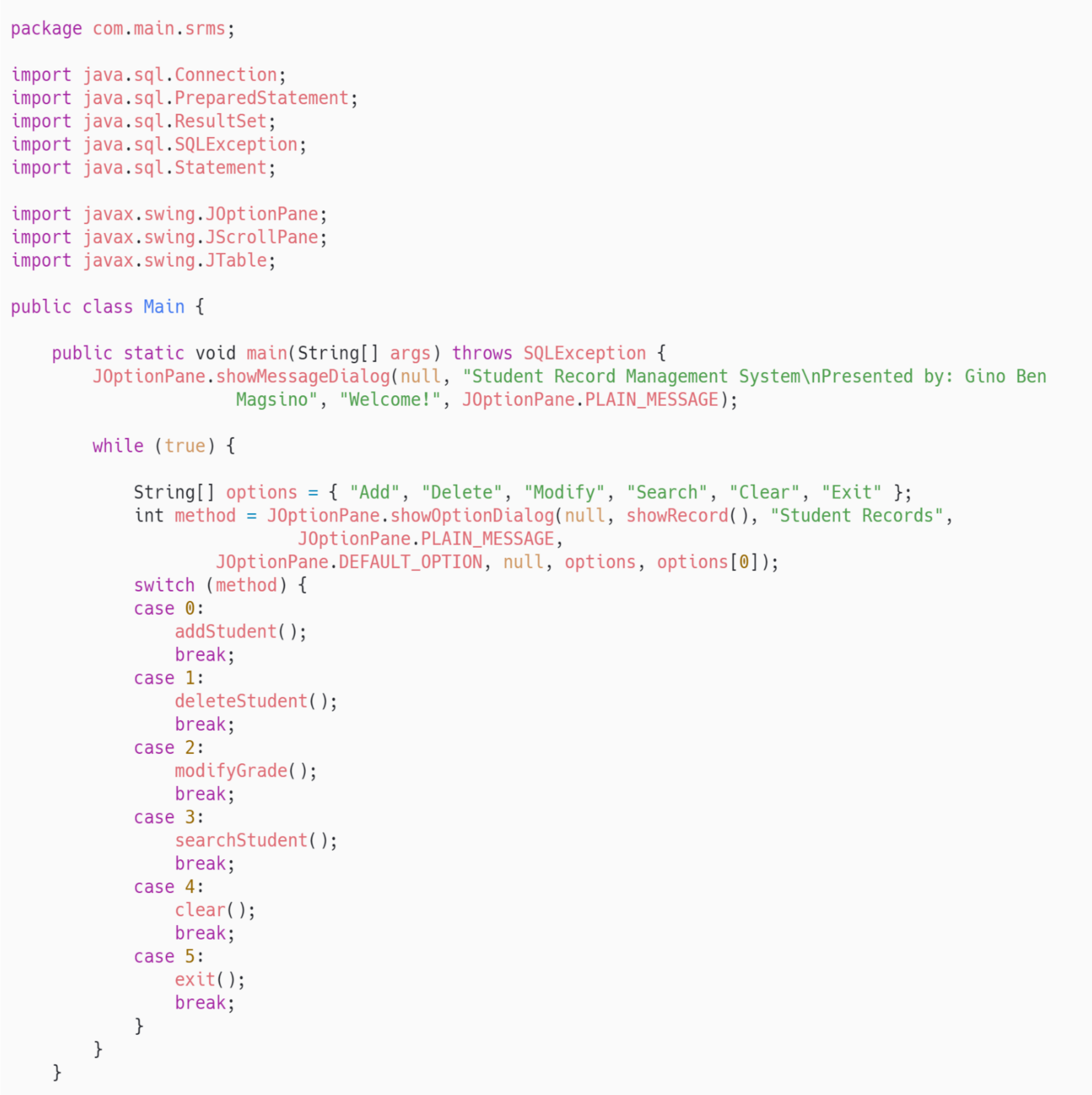




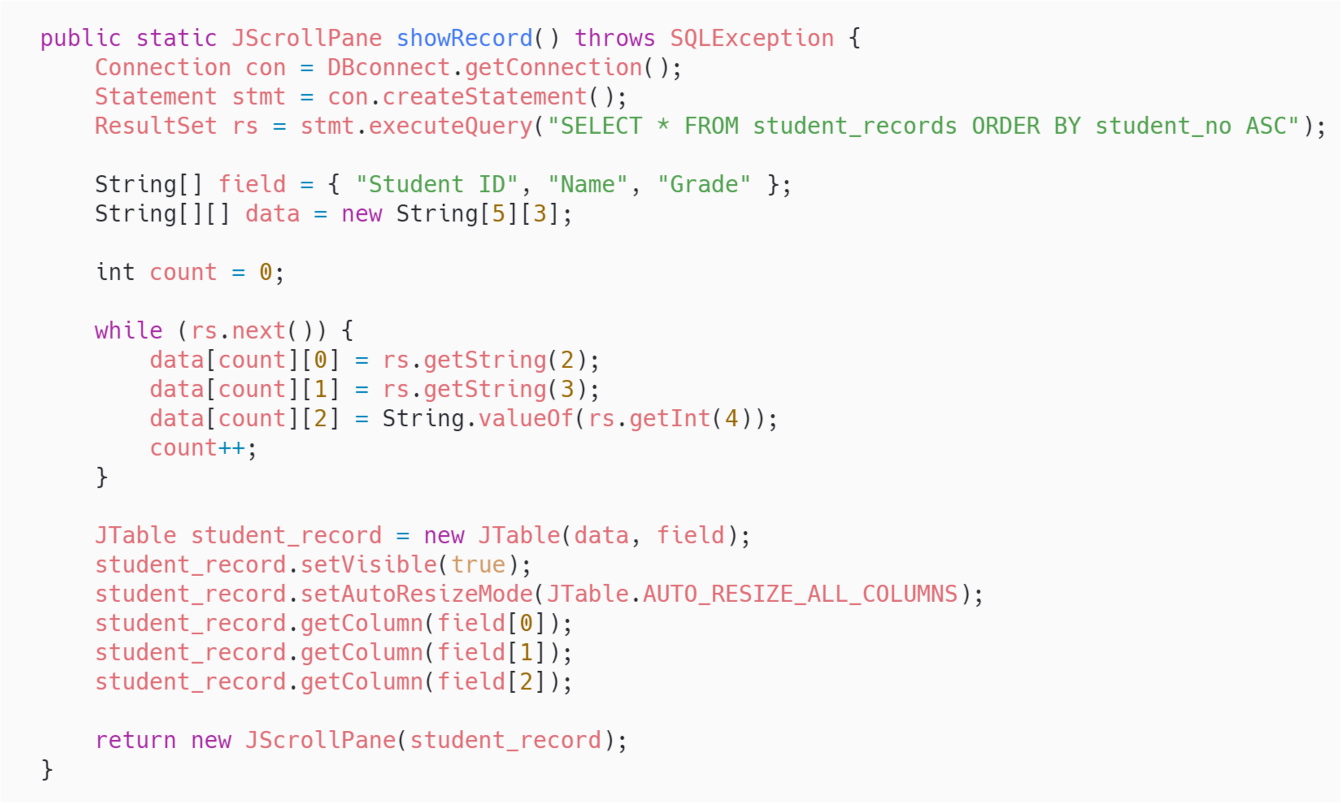
Are you sure?

V. PROGRAM LISTING

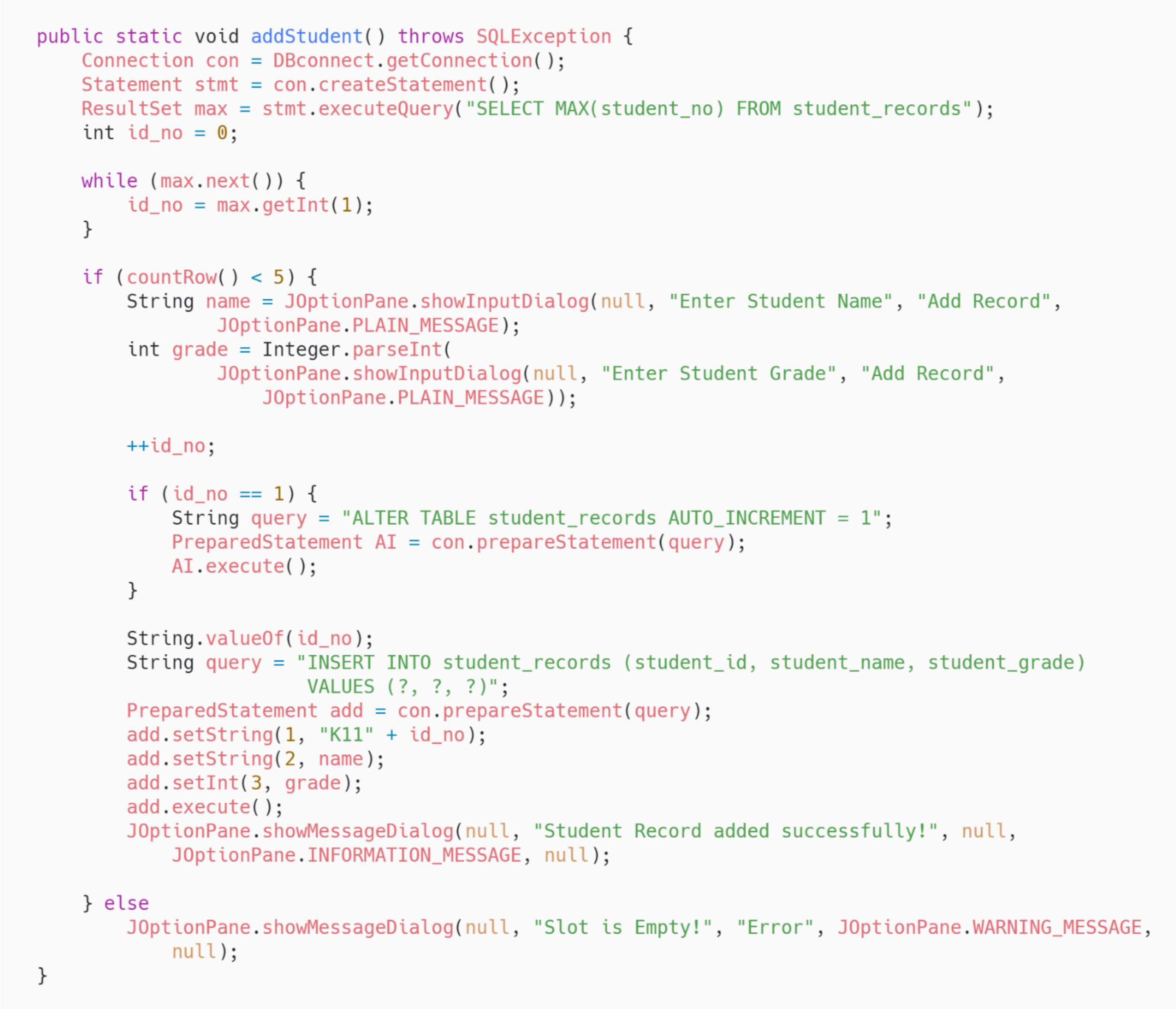
1. Main.class



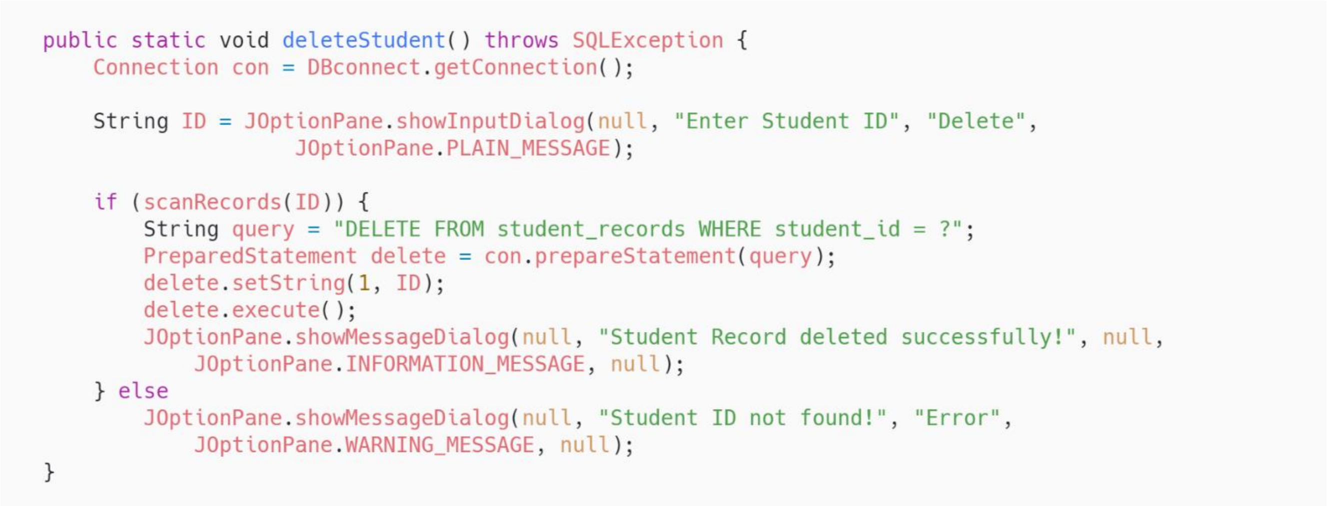
```
package com.main.srms;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

import javax.swing.JOptionPane;
import javax.swing.JScrollPane;
import javax.swing.JTable;

public class Main {

    public static void main(String[] args) throws SQLException {
        JOptionPane.showMessageDialog(null, "Student Record Management System\nPresented by: Gino Ben
                    Magsino", "Welcome!", JOptionPane.PLAIN_MESSAGE);

        while (true) {

            String[] options = { "Add", "Delete", "Modify", "Search", "Clear", "Exit" };
            int method = JOptionPane.showOptionDialog(null, showRecord(), "Student Records",
                            JOptionPane.PLAIN_MESSAGE,
                    JOptionPane.DEFAULT_OPTION, null, options, options[0]);
            switch (method) {
            case 0:
                addStudent();
                break;
            case 1:
                deleteStudent();
                break;
            case 2:
                modifyGrade();
                break;
            case 3:
                searchStudent();
                break;
            case 4:
                clear();
                break;
            case 5:
                exit();
                break;
            }
        }
    }
```



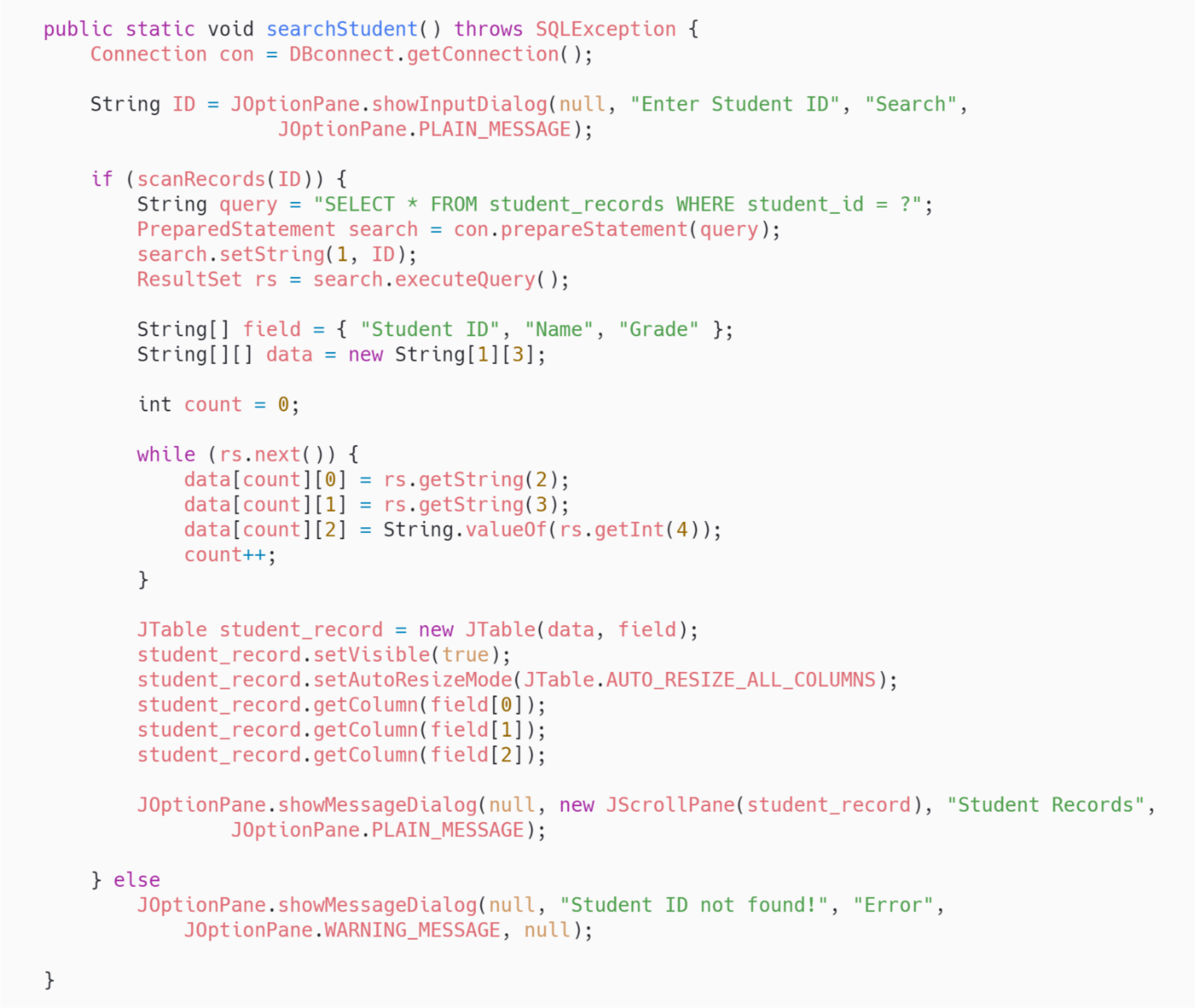
```java
public static JScrollPane showRecord() throws SQLException {
    Connection con = DBconnect.getConnection();
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM student_records ORDER BY student_no ASC");

    String[] field = { "Student ID", "Name", "Grade" };
    String[][] data = new String[5][3];

    int count = 0;

    while (rs.next()) {
        data[count][0] = rs.getString(2);
        data[count][1] = rs.getString(3);
        data[count][2] = String.valueOf(rs.getInt(4));
        count++;
    }

    JTable student_record = new JTable(data, field);
    student_record.setVisible(true);
    student_record.setAutoResizeMode(JTable.AUTO_RESIZE_ALL_COLUMNS);
    student_record.getColumn(field[0]);
    student_record.getColumn(field[1]);
    student_record.getColumn(field[2]);

    return new JScrollPane(student_record);
}
```



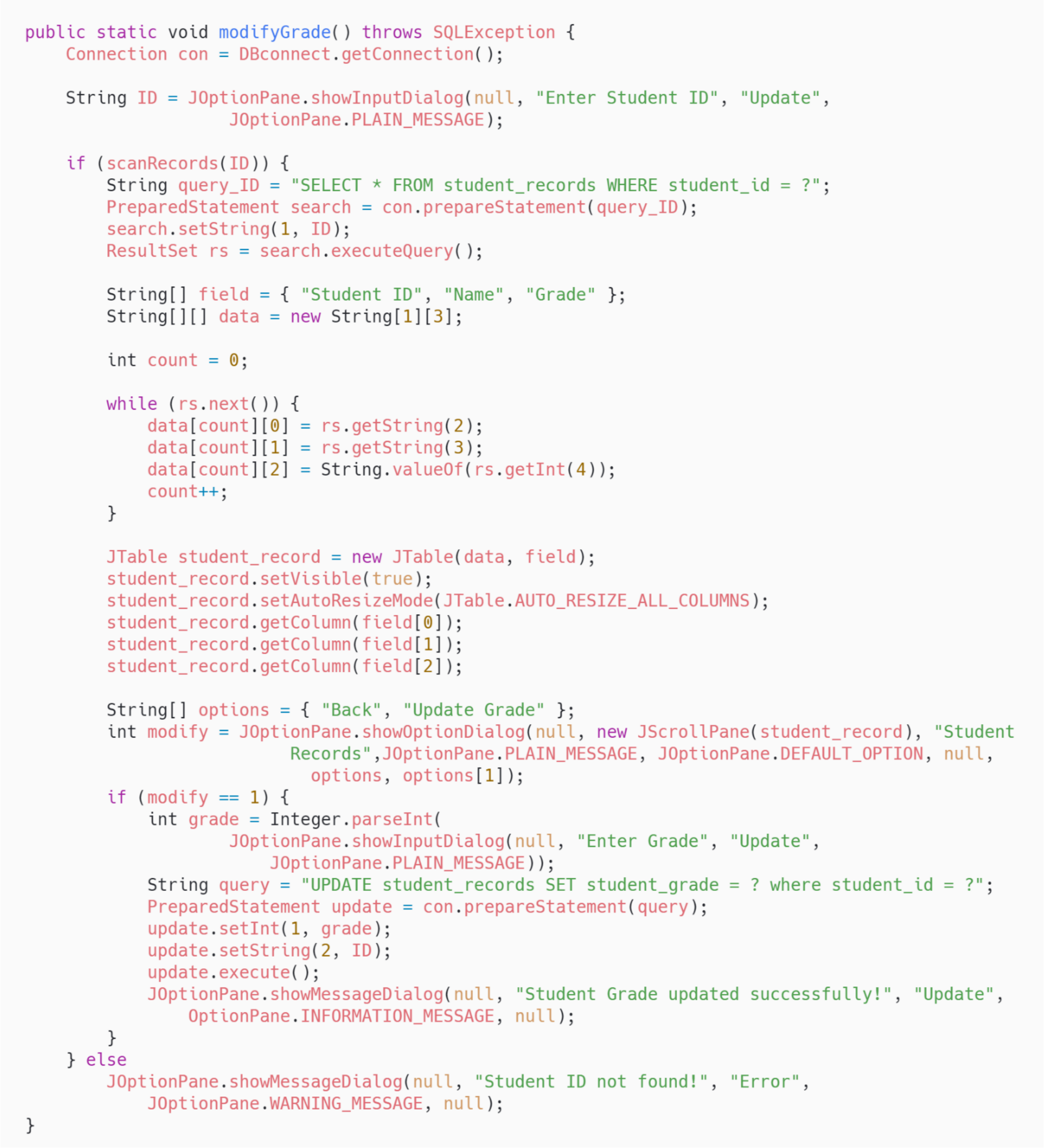
```
public static void addStudent() throws SQLException {
    Connection con = DBconnect.getConnection();
    Statement stmt = con.createStatement();
    ResultSet max = stmt.executeQuery("SELECT MAX(student_no) FROM student_records");
    int id_no = 0;

    while (max.next()) {
        id_no = max.getInt(1);
    }

    if (countRow() < 5) {
        String name = JOptionPane.showInputDialog(null, "Enter Student Name", "Add Record",
                JOptionPane.PLAIN_MESSAGE);
        int grade = Integer.parseInt(
                JOptionPane.showInputDialog(null, "Enter Student Grade", "Add Record",
                    JOptionPane.PLAIN_MESSAGE));

        ++id_no;

        if (id_no == 1) {
            String query = "ALTER TABLE student_records AUTO_INCREMENT = 1";
            PreparedStatement AI = con.prepareStatement(query);
            AI.execute();
        }

        String.valueOf(id_no);
        String query = "INSERT INTO student_records (student_id, student_name, student_grade)
                        VALUES (?, ?, ?)";
        PreparedStatement add = con.prepareStatement(query);
        add.setString(1, "K11" + id_no);
        add.setString(2, name);
        add.setInt(3, grade);
        add.execute();
        JOptionPane.showMessageDialog(null, "Student Record added successfully!", null,
            JOptionPane.INFORMATION_MESSAGE, null);

    } else
        JOptionPane.showMessageDialog(null, "Slot is Empty!", "Error", JOptionPane.WARNING_MESSAGE,
            null);
}
```



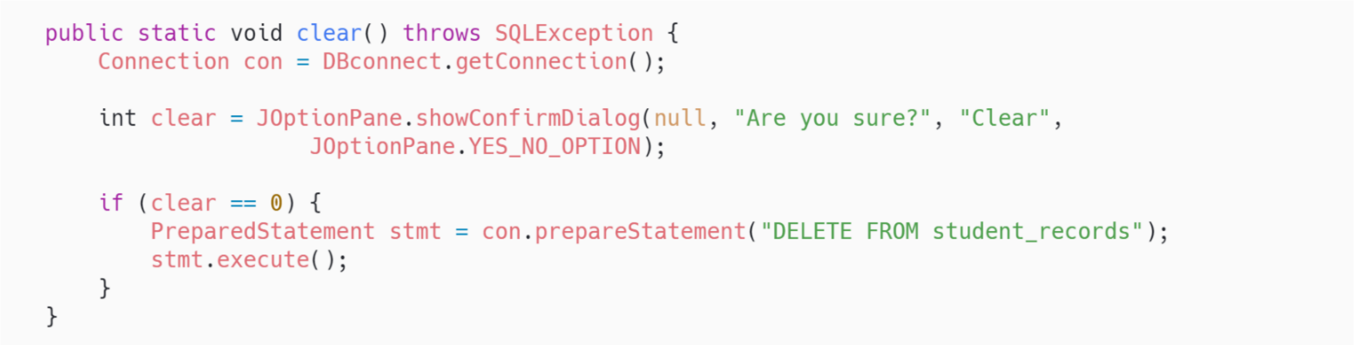
```
public static void deleteStudent() throws SQLException {
    Connection con = DBconnect.getConnection();

    String ID = JOptionPane.showInputDialog(null, "Enter Student ID", "Delete",
                    JOptionPane.PLAIN_MESSAGE);

    if (scanRecords(ID)) {
        String query = "DELETE FROM student_records WHERE student_id = ?";
        PreparedStatement delete = con.prepareStatement(query);
        delete.setString(1, ID);
        delete.execute();
        JOptionPane.showMessageDialog(null, "Student Record deleted successfully!", null,
            JOptionPane.INFORMATION_MESSAGE, null);
    } else
        JOptionPane.showMessageDialog(null, "Student ID not found!", "Error",
            JOptionPane.WARNING_MESSAGE, null);
}
```



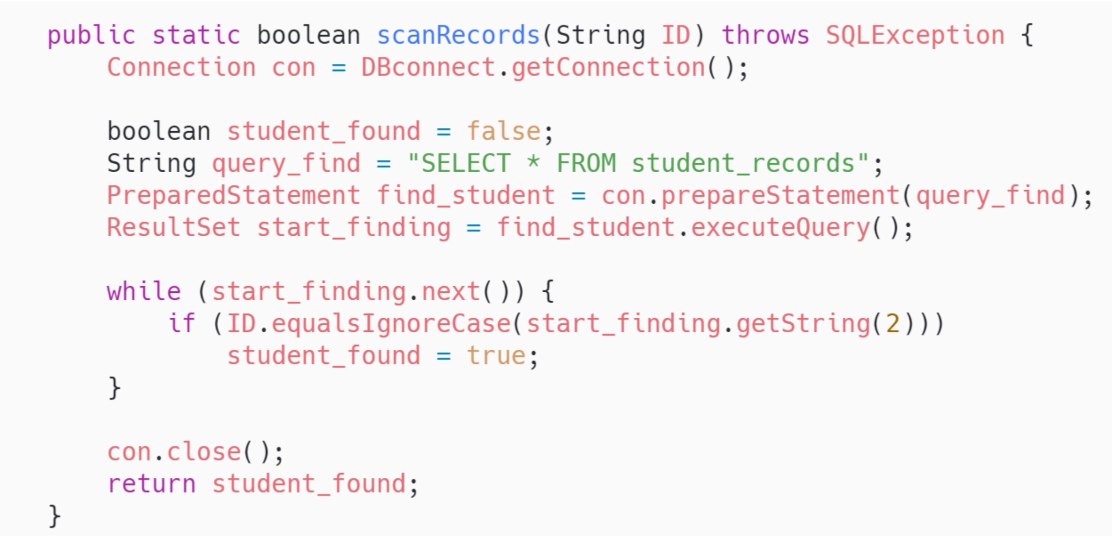
```
public static void searchStudent() throws SQLException {
    Connection con = DBconnect.getConnection();

    String ID = JOptionPane.showInputDialog(null, "Enter Student ID", "Search",
                    JOptionPane.PLAIN_MESSAGE);

    if (scanRecords(ID)) {
        String query = "SELECT * FROM student_records WHERE student_id = ?";
        PreparedStatement search = con.prepareStatement(query);
        search.setString(1, ID);
        ResultSet rs = search.executeQuery();

        String[] field = { "Student ID", "Name", "Grade" };
        String[][] data = new String[1][3];

        int count = 0;

        while (rs.next()) {
            data[count][0] = rs.getString(2);
            data[count][1] = rs.getString(3);
            data[count][2] = String.valueOf(rs.getInt(4));
            count++;
        }

        JTable student_record = new JTable(data, field);
        student_record.setVisible(true);
        student_record.setAutoResizeMode(JTable.AUTO_RESIZE_ALL_COLUMNS);
        student_record.getColumn(field[0]);
        student_record.getColumn(field[1]);
        student_record.getColumn(field[2]);

        JOptionPane.showMessageDialog(null, new JScrollPane(student_record), "Student Records",
                JOptionPane.PLAIN_MESSAGE);

    } else
        JOptionPane.showMessageDialog(null, "Student ID not found!", "Error",
            JOptionPane.WARNING_MESSAGE, null);

}
```



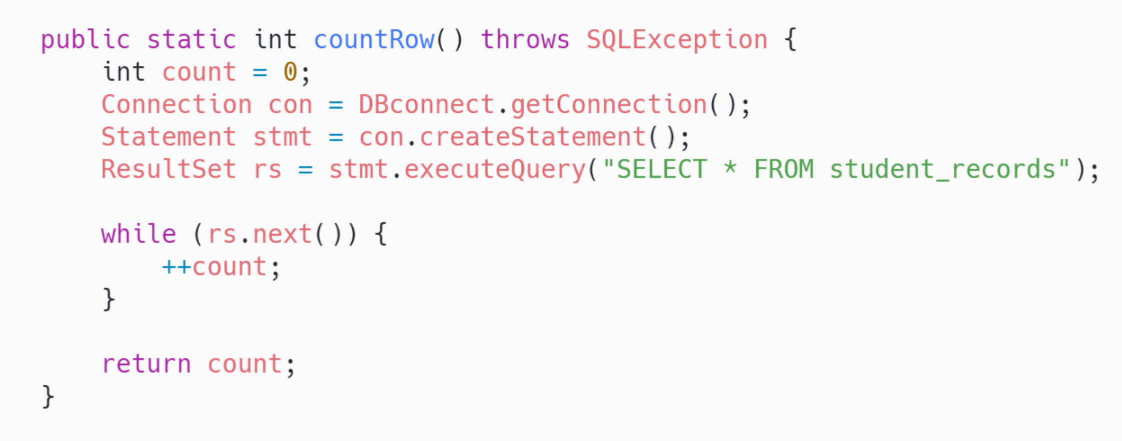
```java
public static void modifyGrade() throws SQLException {
    Connection con = DBconnect.getConnection();

    String ID = JOptionPane.showInputDialog(null, "Enter Student ID", "Update",
                    JOptionPane.PLAIN_MESSAGE);

    if (scanRecords(ID)) {
        String query_ID = "SELECT * FROM student_records WHERE student_id = ?";
        PreparedStatement search = con.prepareStatement(query_ID);
        search.setString(1, ID);
        ResultSet rs = search.executeQuery();

        String[] field = { "Student ID", "Name", "Grade" };
        String[][] data = new String[1][3];

        int count = 0;

        while (rs.next()) {
            data[count][0] = rs.getString(2);
            data[count][1] = rs.getString(3);
            data[count][2] = String.valueOf(rs.getInt(4));
            count++;
        }

        JTable student_record = new JTable(data, field);
        student_record.setVisible(true);
        student_record.setAutoResizeMode(JTable.AUTO_RESIZE_ALL_COLUMNS);
        student_record.getColumn(field[0]);
        student_record.getColumn(field[1]);
        student_record.getColumn(field[2]);

        String[] options = { "Back", "Update Grade" };
        int modify = JOptionPane.showOptionDialog(null, new JScrollPane(student_record), "Student
                          Records",JOptionPane.PLAIN_MESSAGE, JOptionPane.DEFAULT_OPTION, null,
                            options, options[1]);
        if (modify == 1) {
            int grade = Integer.parseInt(
                    JOptionPane.showInputDialog(null, "Enter Grade", "Update",
                        JOptionPane.PLAIN_MESSAGE));
            String query = "UPDATE student_records SET student_grade = ? where student_id = ?";
            PreparedStatement update = con.prepareStatement(query);
            update.setInt(1, grade);
            update.setString(2, ID);
            update.execute();
            JOptionPane.showMessageDialog(null, "Student Grade updated successfully!", "Update",
                OptionPane.INFORMATION_MESSAGE, null);
        }
    } else
        JOptionPane.showMessageDialog(null, "Student ID not found!", "Error",
            JOptionPane.WARNING_MESSAGE, null);
}
```



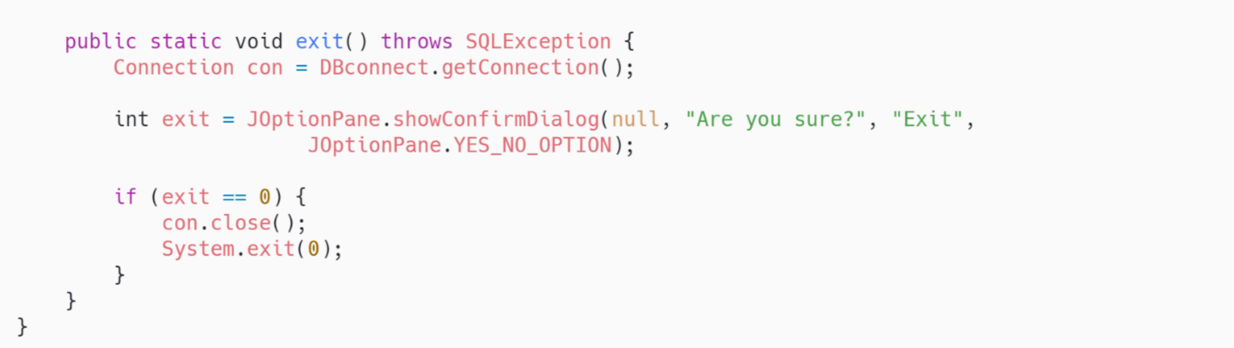
```
public static void clear() throws SQLException {
    Connection con = DBconnect.getConnection();

    int clear = JOptionPane.showConfirmDialog(null, "Are you sure?", "Clear",
                JOptionPane.YES_NO_OPTION);

    if (clear == 0) {
        PreparedStatement stmt = con.prepareStatement("DELETE FROM student_records");
        stmt.execute();
    }
}
```



```
public static boolean scanRecords(String ID) throws SQLException {
    Connection con = DBconnect.getConnection();

    boolean student_found = false;
    String query_find = "SELECT * FROM student_records";
    PreparedStatement find_student = con.prepareStatement(query_find);
    ResultSet start_finding = find_student.executeQuery();

    while (start_finding.next()) {
        if (ID.equalsIgnoreCase(start_finding.getString(2)))
            student_found = true;
    }

    con.close();
    return student_found;
}
```

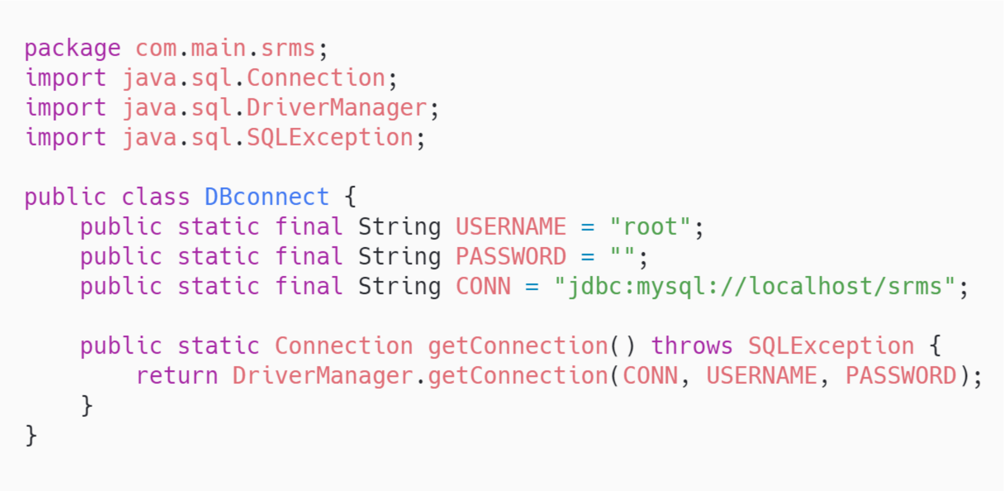


```
public static int countRow() throws SQLException {
    int count = 0;
    Connection con = DBconnect.getConnection();
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM student_records");

    while (rs.next()) {
        ++count;
    }

    return count;
}
```



```
    public static void exit() throws SQLException {
        Connection con = DBconnect.getConnection();

        int exit = JOptionPane.showConfirmDialog(null, "Are you sure?", "Exit",
                        JOptionPane.YES_NO_OPTION);

        if (exit == 0) {
            con.close();
            System.exit(0);
        }
    }
}
```

2. DBConnect.class



```
package com.main.srms;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBconnect {
    public static final String USERNAME = "root";
    public static final String PASSWORD = "";
    public static final String CONN = "jdbc:mysql://localhost/srms";

    public static Connection getConnection() throws SQLException {
        return DriverManager.getConnection(CONN, USERNAME, PASSWORD);
    }
}
```