

VISUAL PROGRAMMING PROJECT

DOCUMENTATION STAGE 4

Project Name : Student Information System

Student ID: 1031510030

Student Name: Büşra Yıldırım

1. Project Definition and Purpose

This project is a Student and Teacher Management System designed to simplify academic processes in Highschool environment. It includes features like managing students, courses, and grades, as well as generating PDF reports based on specific criteria. The system provides separate interfaces for teachers and students, making it easy for teachers to add courses, record grades, track attendance, and generate detailed reports. Students can view their grades, attendance records, and transcripts, with the option to export these as PDF files.

First of all, in this project, we created 1 teacher and 10 students as default in a table in our database and assigned a password to each of them. and the events performed for each user are recorded instantly in the database.

Teacher and Student Functionalities

Teacher Functionalities

1-) Course Management:

Teachers can create, update, or remove courses. These courses are stored in the database and linked with students and their grades.

Course Management

Course Details

Course Name:

Instructor:

Credits:

Add Course

Update Course

Delete Course

Course Name	Instructor	Credits
Math	Osman Mucuk	6
Physics	Emrah Tiras	6
English	Ayşe Demir	4
P.E	Ali Can	2
Biology	Havva Aktürk	3
Chemistry	Şeyma Temiz	3
Turkish	Ayşe Koçak	6

Back

```
public static void createCourseTable() {
    String sql = "CREATE TABLE IF NOT EXISTS courses ("
        + "id INTEGER PRIMARY KEY AUTOINCREMENT, "
        + "course_name TEXT NOT NULL"
        + ");";
    try (Connection conn = connect(); Statement stmt = conn.createStatement()) {
        stmt.execute(sql);
        System.out.println("Courses table created.");
    } catch (SQLException e) {
        System.out.println("Error creating courses table: " + e.getMessage());
    }
}
```

```
CREATE TABLE courses (
    id            INTEGER PRIMARY KEY AUTOINCREMENT,
    course_name   TEXT    NOT NULL,
    instructor    TEXT    NOT NULL,
    credits       INTEGER NOT NULL
);
```

2-)Grade Management:

Teachers can add, update, or delete grades for students in specific courses. This allows for seamless grading and keeps all records up-to-date in the database.

Student Management

Back

ID	Name	Surname	Class
1	Ayşe	Yılmaz	9A
2	Ali	Can	10B
3	Fatma	Demir	11C
4	Mehmet	Kaya	12A
5	Zeynep	Şahin	9B
6	Mustafa	Çelik	10A
7	Elif	Öztürk	11B
8	Emre	Yıldız	12B
9	Selin	Arslan	9C
10	Oğuz	Doğan	10C

Math

Add Grade

Delete Grade


Update Grade

Student ID	Course	Grade
1	Math	100
2	Physics	99
3	English	55
4	P.E	66
5	English	80

```
CREATE TABLE grades (  
    id          INTEGER PRIMARY KEY AUTOINCREMENT,  
    student_id  INTEGER NOT NULL,  
    course      TEXT     NOT NULL,  
    grade       INTEGER NOT NULL,  
    FOREIGN KEY (  
        student_id  
    )  
    REFERENCES students (id)  
);
```

3-)Attendance Management:

Teachers can record and track student attendance. Attendance data is stored in the database and can be used for analysis or reporting.

 Attendance Management

Back

Student ID	Date
1	2024-12-18
1	2025-12-21
1	2025-12-13
2	2026-12-18
6	2026-12-22
65654	2024-12-19

Student ID:

Date:

▲
▼

Add Attendance

Delete Attendance

```
CREATE TABLE attendance (  
    id          INTEGER PRIMARY KEY AUTOINCREMENT,  
    student_id  INTEGER NOT NULL,  
    date        TEXT     NOT NULL,  
    FOREIGN KEY (  
        student_id  
    )  
    REFERENCES students (id)  
);
```

```
public static void addAttendance(int studentID, String date) {  
    String sql = "INSERT INTO attendance (student_id, date) VALUES (?, ?)";  
    try (Connection conn = connect(); PreparedStatement pstmt = conn.prepareStatement(sql))  
    {  
        pstmt.setInt(1, studentID);  
        pstmt.setString(2, date);  
        pstmt.executeUpdate();  
    } catch (SQLException e) {  
        System.out.println("Error adding attendance: " + e.getMessage());  
    }  
}
```

4-)Generate Customized Reports:

Teachers can generate PDF reports based on specific criteria, such as:

- Grades for a particular student. Grades for a specific course.
- Attendance records for selected students or entire classes. These reports can be exported for record-keeping, printing, or sharing.
- The iText library is used for generating PDF reports, ensuring portability and accessibility of data.

Teacher Report

Back

Export to PDF

Course:

Select a course

☐ All Courses

Student ID:

☐ All Students

Student ID	Name	Surname	Class	Course	Grade
1	Ayşe	Yılmaz	9A	Math	100
2	Ali	Can	10B	Physics	99
3	Fatma	Demir	11C	English	55
4	Mehmet	Kaya	12A	P.E	66
5	Zeynep	Şahin	9B	English	80
6	Mustafa	Çelik	10A	Chemistry	88
7	Elif	Öztürk	11B	P.E	78
8	Emre	Yıldız	12B	Biology	47
9	Selin	Arslan	9C	Math	69
10	Oğuz	Doğan	10C	Physics	64
1	Ayşe	Yılmaz	9A	Chemistry	98
1	Ayşe	Yılmaz	9A	Biology	74
1	Ayşe	Yılmaz	9A	Physics	54
1	Ayşe	Yılmaz	9A	English	75

TeacherReport.pdf

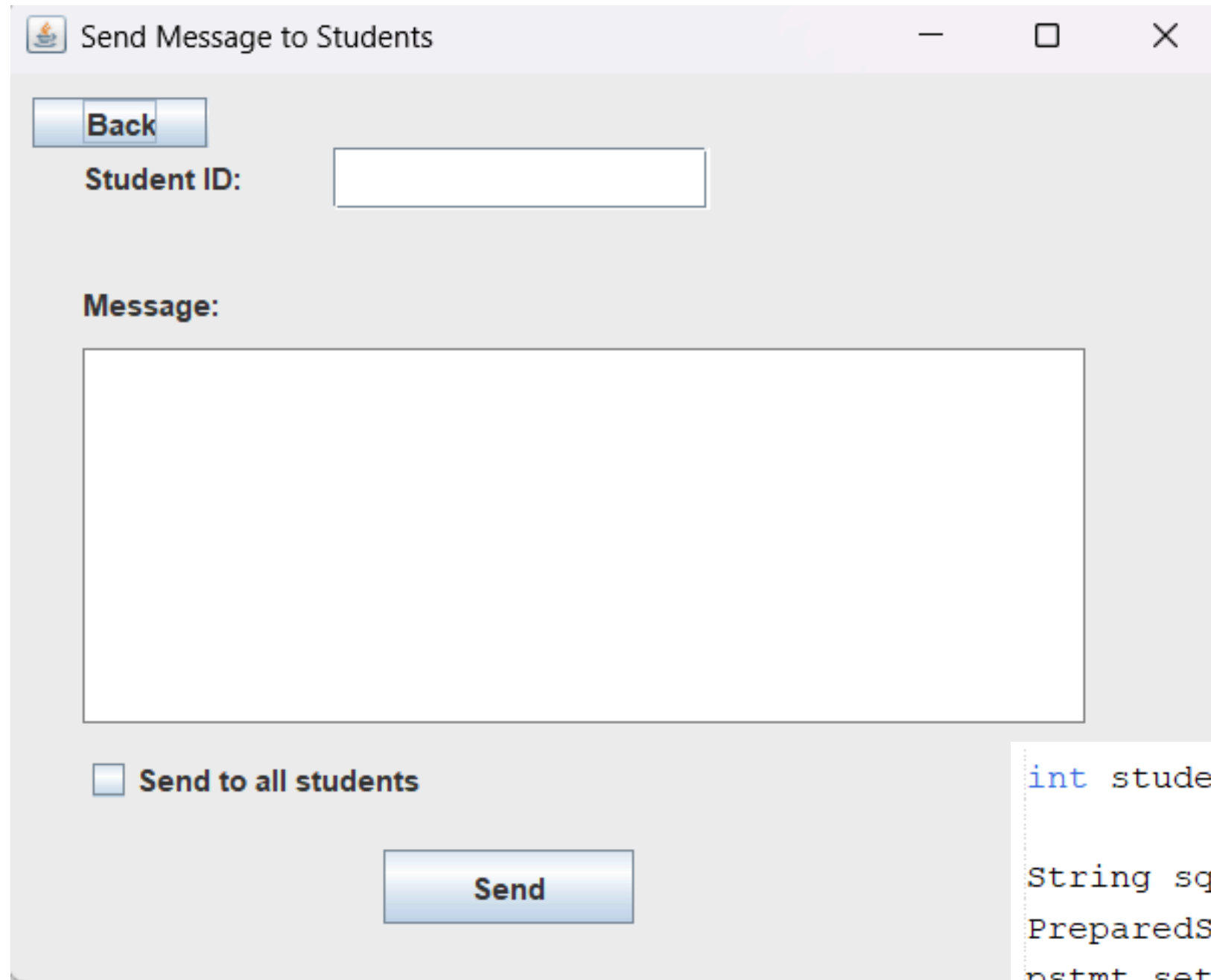
Dosya | C:/Users/Hp/Downloads/TeacherReport.pdf

Teacher Report

Student ID	Name	Surname	Class	Course	Grade
1	Aye	Ylmaz	9A	Math	100
2	Ali	Can	10B	Physics	99
3	Fatma	Demir	11C	English	55
4	Mehmet	Kaya	12A	P.E	66
5	Zeynep	ahin	9B	English	80
6	Mustafa	Çelik	10A	Chemistry	88
7	Elif	Öztürk	11B	P.E	78
8	Emre	Yldz	12B	Biology	47
9	Selin	Arslan	9C	Math	69
10	Ouz	Doan	10C	Physics	64
1	Aye	Ylmaz	9A	Chemistry	98
1	Aye	Ylmaz	9A	Biology	74
1	Aye	Ylmaz	9A	Physics	54
1	Aye	Ylmaz	9A	English	75

5-) Teacher message screen

The teacher can send a message to any student (by entering their ID). and this message will be displayed on the student's page. He/she can select all students and send the message to everyone.



Send Message to Students

Student ID:

Message:

☐ Send to all students

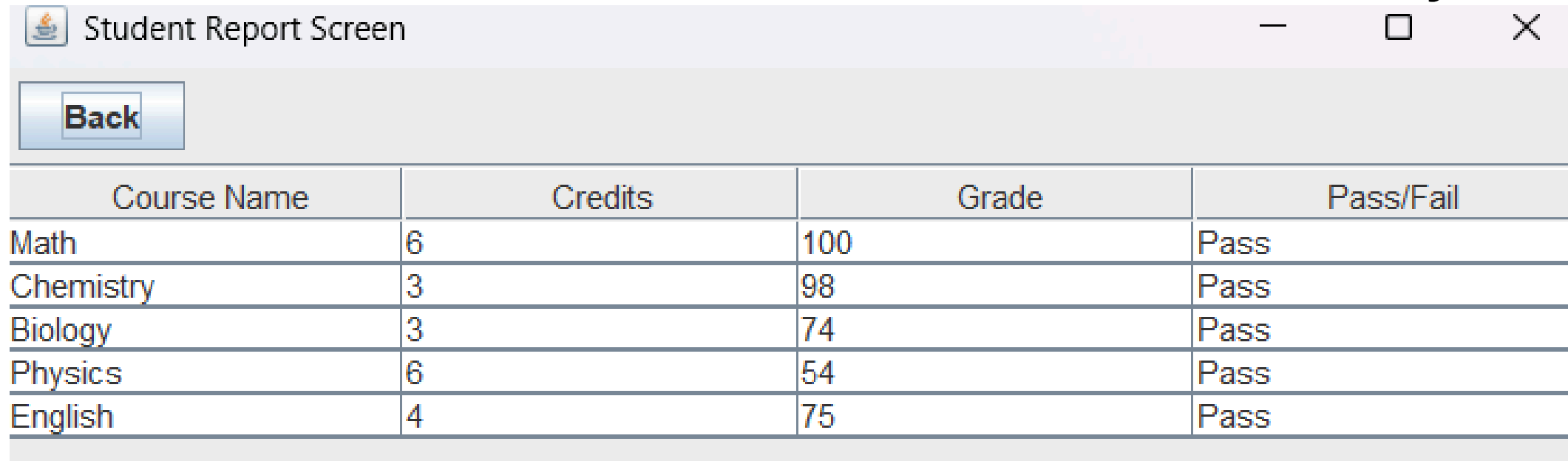
```
CREATE TABLE messages (  
    id          INTEGER PRIMARY KEY AUTOINCREMENT,  
    student_id  INTEGER,  
    message     TEXT      NOT NULL,  
    sender      TEXT      NOT NULL  
                DEFAULT 'Teacher',  
    timestamp   DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

```
int studentID = rs.getInt("id");  
  
String sql = "INSERT INTO messages (student_id, message, sender) VALUES (?, ?, ?)";  
PreparedStatement pstmt = conn.prepareStatement(sql);  
pstmt.setInt(1, studentID);  
pstmt.setString(2, message);  
pstmt.setString(3, "Teacher");  
pstmt.executeUpdate();
```


Student Functionalities

1-)View Grade Reports:

Students can view their grades for all completed courses. These grades are pulled directly from the database for real-time accuracy.



The screenshot shows a window titled "Student Report Screen" with a "Back" button and a table of student grades. The table has four columns: Course Name, Credits, Grade, and Pass/Fail. The data rows are as follows:

Course Name	Credits	Grade	Pass/Fail
Math	6	100	Pass
Chemistry	3	98	Pass
Biology	3	74	Pass
Physics	6	54	Pass
English	4	75	Pass

```
public static List<String[]> getStudentGrades() {  
    List<String[]> studentGrades = new ArrayList<>();  
    String sql = "SELECT s.id AS student_id, s.name, s.surname, s.class, g.course, g.grade "  
        + "FROM students s "  
        + "INNER JOIN grades g ON s.id = g.student_id";  
  
    try (Connection conn = connect(); Statement stmt = conn.createStatement(); ResultSet rs = stmt.executeQuery(sql)) {  
  
        while (rs.next()) {  
            String[] row = {  
                rs.getString("student_id"),  
                rs.getString("name"),  
                rs.getString("surname"),  
                rs.getString("class"),  
                rs.getString("course"),  
                rs.getString("grade")  
            };  
            studentGrades.add(row);  
        }  
    }  
}
```


3-)Access Transcripts:

Students can generate their academic transcripts in PDF format. These transcripts summarize their performance across all courses. The iText library is used for generating PDF reports, ensuring portability and accessibility of data.

Transcript

Back

Generate PDF

Student Information

Student Info

Student ID: 1

Department: Placeholder Department

Course Na...	Credits	Grade	Total Points	Pass/Fail
Math	6	100	600	Pass
Chemistry	3	98	294	Pass
Biology	3	74	222	Pass
Physics	6	54	324	Pass
English	4	75	300	Pass

GPA: 79.09

Transcript_1.pdf

Dosya | C:/Users/Hp/Downloads/Transcript_1.pdf

1 / 1

Acrobat ile düz

Student Transcript

Student ID: 1

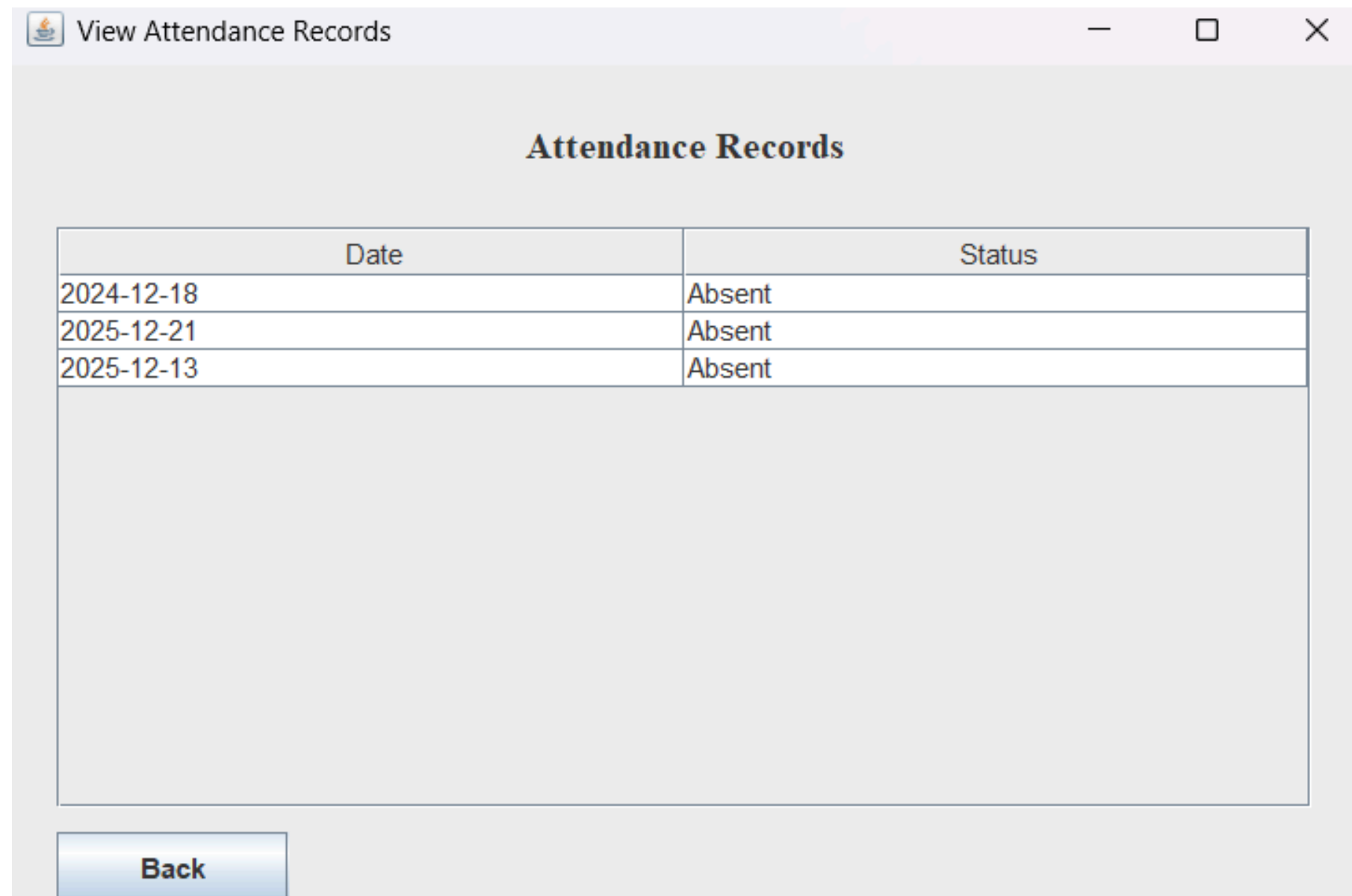
Department: Placeholder Department

Course Name	Credits	Grade	Total Points	Pass/Fail
Math	6	100	600	Pass
Chemistry	3	98	294	Pass
Biology	3	74	222	Pass
Physics	6	54	324	Pass
English	4	75	300	Pass

GPA: 79.09

3-)Attendance Tracking:

- Students can view their attendance records to monitor their participation in classes.



View Attendance Records

Attendance Records

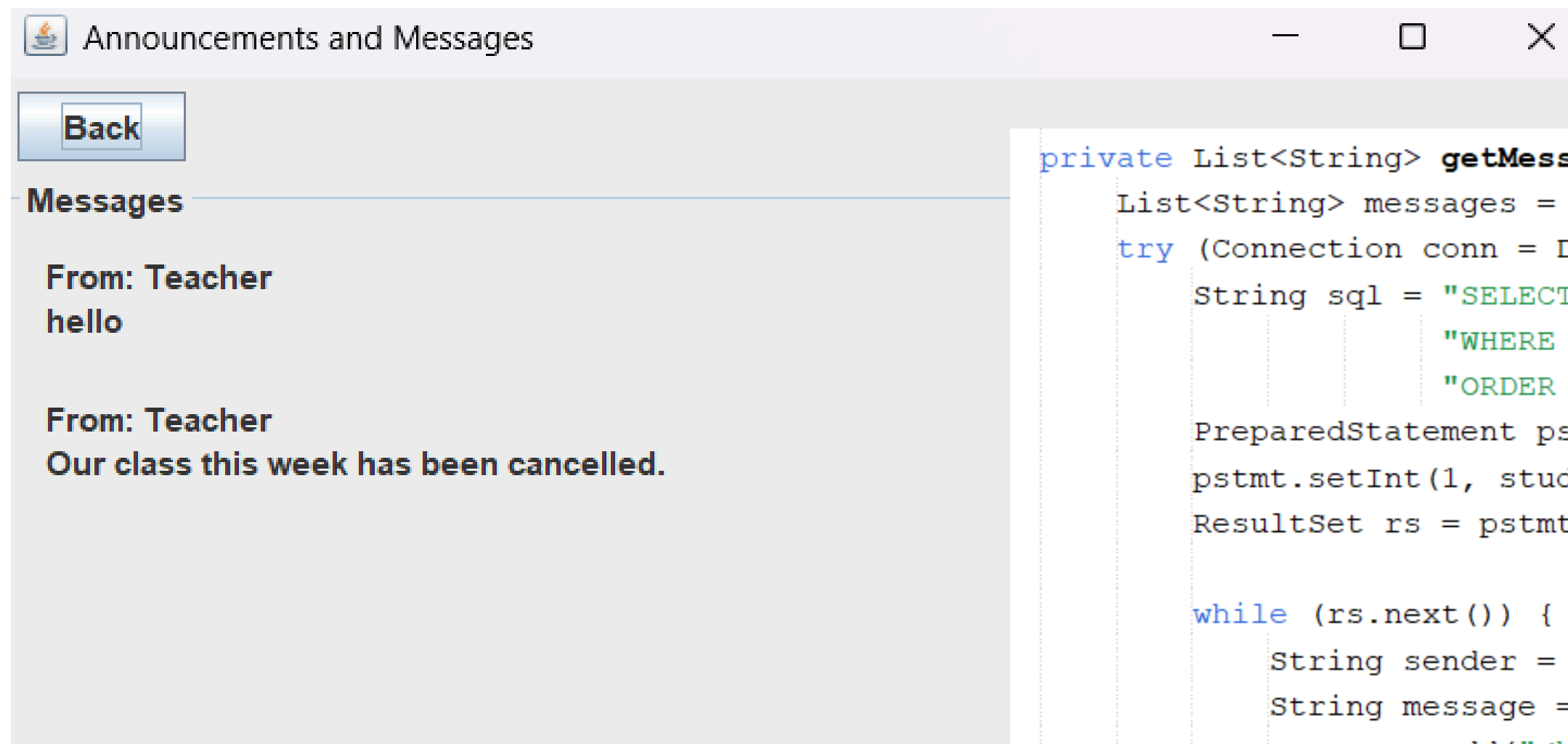
Date	Status
2024-12-18	Absent
2025-12-21	Absent
2025-12-13	Absent

Back

```
CREATE TABLE attendance (  
    id          INTEGER PRIMARY KEY AUTOINCREMENT,  
    student_id  INTEGER NOT NULL,  
    date        TEXT     NOT NULL,  
    FOREIGN KEY (  
        student_id  
    )  
    REFERENCES students (id)  
);
```

4-) Announcements And Messages

In the message and announcements section, our student can view the message sent to him by the teacher. This message is stored in the message section of the database.



```
private List<String> getMessagesForStudent(int studentId) {  
    List<String> messages = new ArrayList<>();  
    try (Connection conn = DatabaseHelper.connect()) {  
        String sql = "SELECT sender, message FROM messages " +  
            "WHERE student_id = ? OR student_id IS NULL " +  
            "ORDER BY timestamp DESC";  
        PreparedStatement pstmt = conn.prepareStatement(sql);  
        pstmt.setInt(1, studentId);  
        ResultSet rs = pstmt.executeQuery();  
  
        while (rs.next()) {  
            String sender = rs.getString("sender");  
            String message = rs.getString("message");  
            messages.add("<html><b>From: " + sender + "</b><br>" + message + "</html>");  
        }  
    } catch (SQLException e) {  
        e.printStackTrace();  
    }  
    return messages;  
}
```

Thank You For Listening... :)