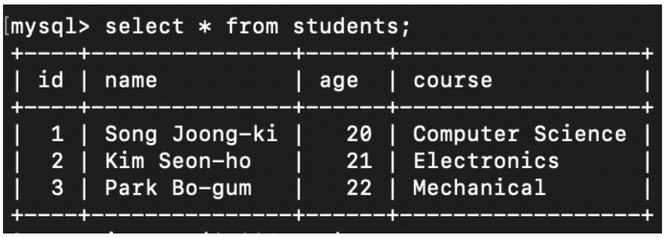
Roll no. 31145

Assignment 8

MySQL part:

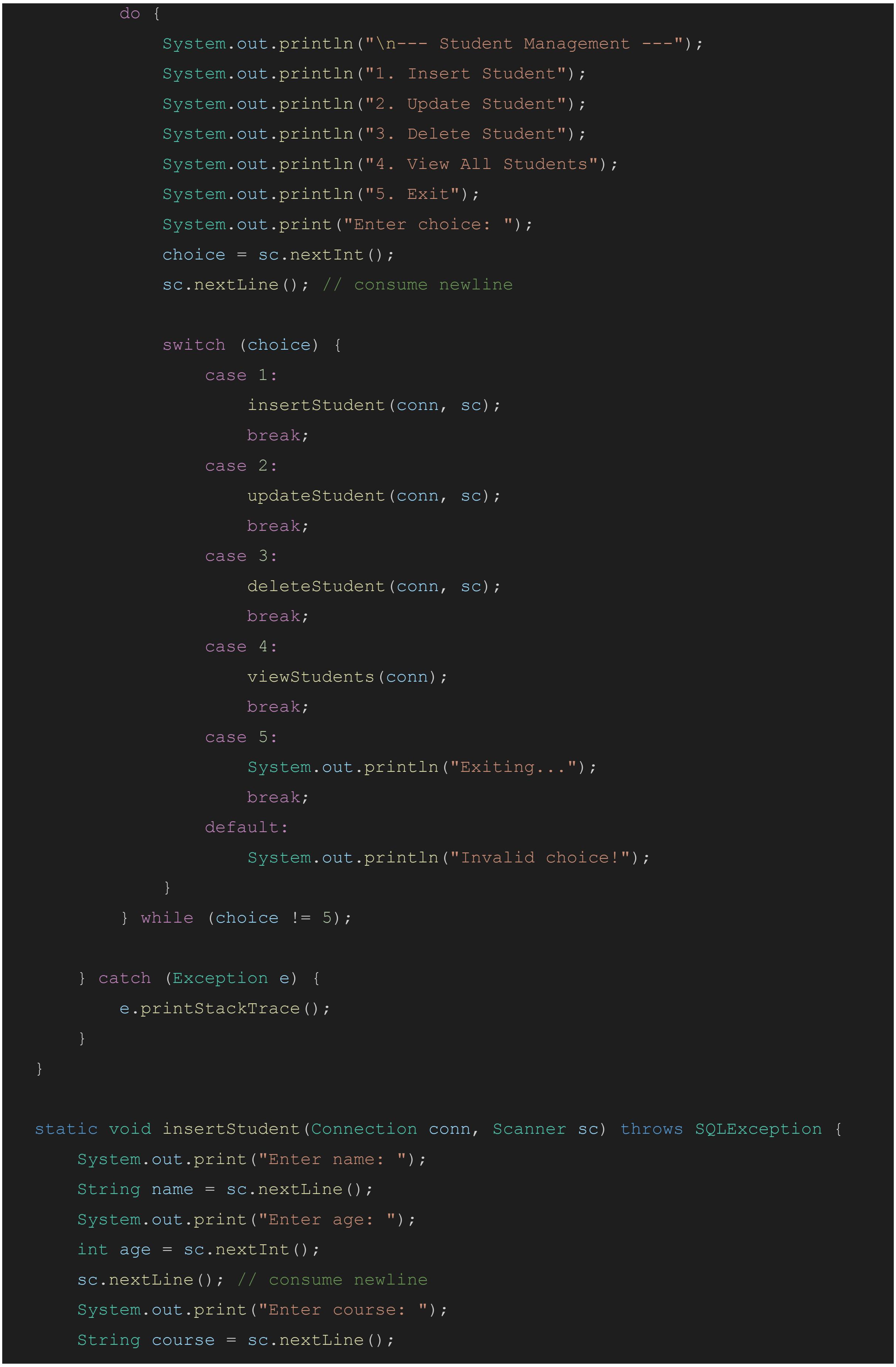
```
CREATE TABLE students (
    id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(50),
    age INT,
    course VARCHAR(50)
);

INSERT INTO students (name, age, course)
VALUES
  ('Song Joong-ki', 20, 'Computer Science'),
  ('Kim Seon-ho', 21, 'Electronics'),
  ('Park Bo-gum', 22, 'Mechanical');

SELECT * FROM students;
```



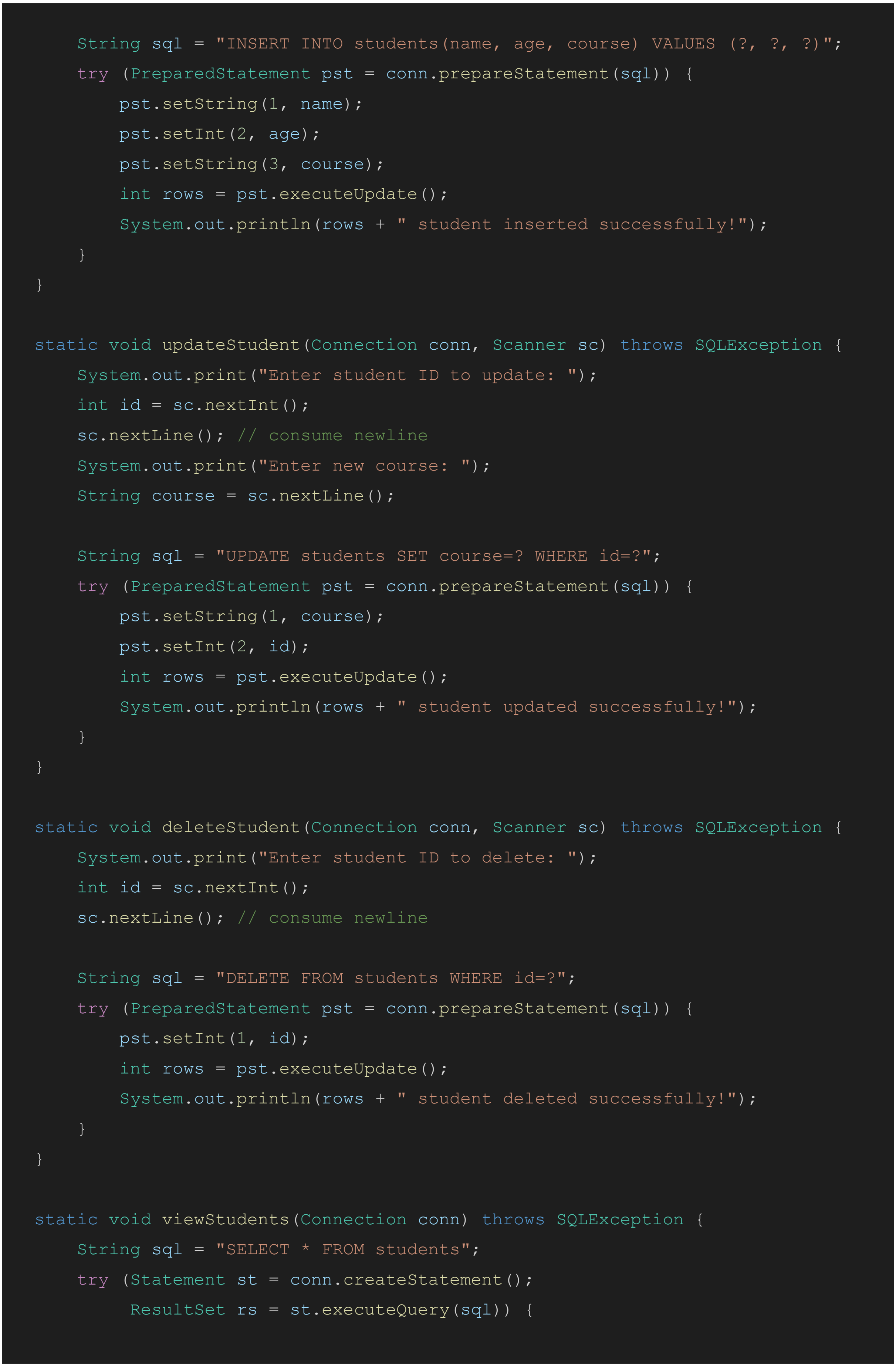
```
mysql> select * from students;
+----+---------------+------+------------------+
| id | name          | age  | course           |
+----+---------------+------+------------------+
|  1 | Song Joong-ki |   20 | Computer Science |
|  2 | Kim Seon-ho   |   21 | Electronics      |
|  3 | Park Bo-gum   |   22 | Mechanical       |
+----+---------------+------+------------------+
```

Java part:

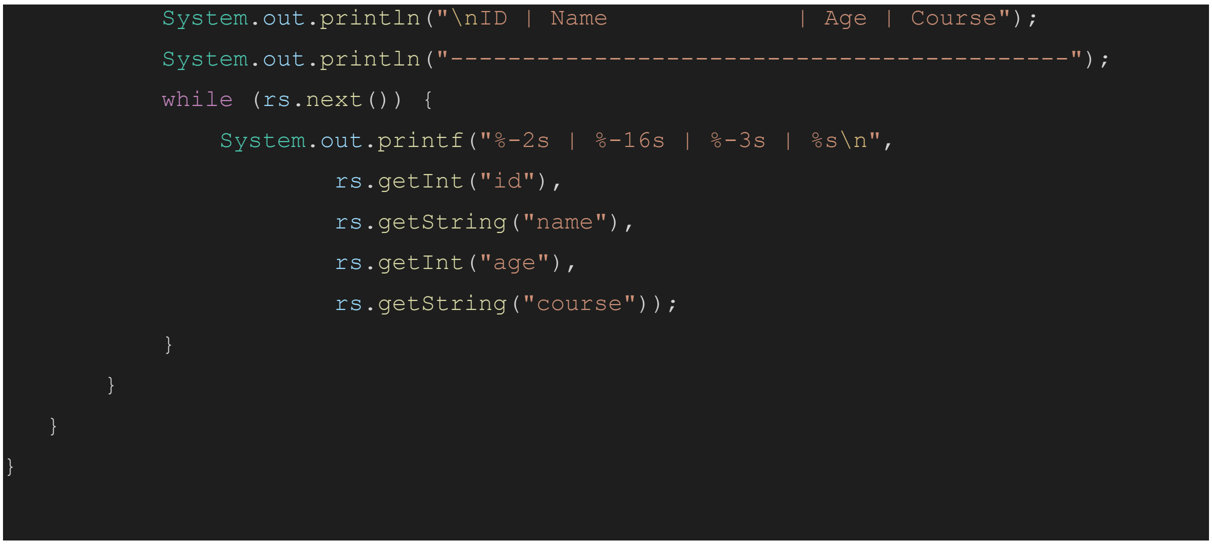
```java
import java.sql.*;
import java.util.Scanner;

public class Main {
    static final String URL =
"jdbc:mysql://localhost:3306/dbms_sem5?useSSL=false&serverTimezone=UTC";
    static final String USER = "root";           // your MySQL user
    static final String PASSWORD = "2312";       // your MySQL password

    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(URL, USER, PASSWORD);
             Scanner sc = new Scanner(System.in)) {

            Class.forName("com.mysql.cj.jdbc.Driver"); // Load driver
            System.out.println("✅ Connected to dbms_sem5 database!");

            int choice = 0;
```



```
            do {
                System.out.println("\n--- Student Management ---");
                System.out.println("1. Insert Student");
                System.out.println("2. Update Student");
                System.out.println("3. Delete Student");
                System.out.println("4. View All Students");
                System.out.println("5. Exit");
                System.out.print("Enter choice: ");
                choice = sc.nextInt();
                sc.nextLine(); // consume newline

                switch (choice) {
                    case 1:
                        insertStudent(conn, sc);
                        break;
                    case 2:
                        updateStudent(conn, sc);
                        break;
                    case 3:
                        deleteStudent(conn, sc);
                        break;
                    case 4:
                        viewStudents(conn);
                        break;
                    case 5:
                        System.out.println("Exiting...");
                        break;
                    default:
                        System.out.println("Invalid choice!");
                }
            } while (choice != 5);

        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    static void insertStudent(Connection conn, Scanner sc) throws SQLException {
        System.out.print("Enter name: ");
        String name = sc.nextLine();
        System.out.print("Enter age: ");
        int age = sc.nextInt();
        sc.nextLine(); // consume newline
        System.out.print("Enter course: ");
        String course = sc.nextLine();
```



```java
    String sql = "INSERT INTO students(name, age, course) VALUES (?, ?, ?)";
    try (PreparedStatement pst = conn.prepareStatement(sql)) {
        pst.setString(1, name);
        pst.setInt(2, age);
        pst.setString(3, course);
        int rows = pst.executeUpdate();
        System.out.println(rows + " student inserted successfully!");
    }
}

static void updateStudent(Connection conn, Scanner sc) throws SQLException {
    System.out.print("Enter student ID to update: ");
    int id = sc.nextInt();
    sc.nextLine(); // consume newline
    System.out.print("Enter new course: ");
    String course = sc.nextLine();

    String sql = "UPDATE students SET course=? WHERE id=?";
    try (PreparedStatement pst = conn.prepareStatement(sql)) {
        pst.setString(1, course);
        pst.setInt(2, id);
        int rows = pst.executeUpdate();
        System.out.println(rows + " student updated successfully!");
    }
}

static void deleteStudent(Connection conn, Scanner sc) throws SQLException {
    System.out.print("Enter student ID to delete: ");
    int id = sc.nextInt();
    sc.nextLine(); // consume newline

    String sql = "DELETE FROM students WHERE id=?";
    try (PreparedStatement pst = conn.prepareStatement(sql)) {
        pst.setInt(1, id);
        int rows = pst.executeUpdate();
        System.out.println(rows + " student deleted successfully!");
    }
}

static void viewStudents(Connection conn) throws SQLException {
    String sql = "SELECT * FROM students";
    try (Statement st = conn.createStatement();
         ResultSet rs = st.executeQuery(sql)) {
```



```java
            System.out.println("\nID | Name              | Age | Course");
            System.out.println("-------------------------------------------");
            while (rs.next()) {
                System.out.printf("%-2s | %-16s | %-3s | %s\n",
                        rs.getInt("id"),
                        rs.getString("name"),
                        rs.getInt("age"),
                        rs.getString("course"));
            }
        }
    }
}
```

Test cases:

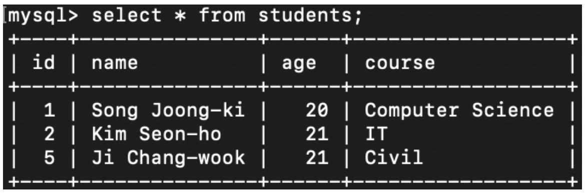
```
riddhilahare@Riddhis-MacBook-Air assig8 % javac -cp ".:lib/mysql-connector-j-9.4.0.jar"
src/Main.java -d out
riddhilahare@Riddhis-MacBook-Air assig8 % java -cp ".:out:lib/mysql-connector-j-9.4.0.jar"
Main
✅ Connected to dbms_sem5 database!

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 4

ID | Name              | Age | Course
-------------------------------------------
1  | Song Joong-ki     | 20  | Computer Science
2  | Kim Seon-ho       | 21  | Electronics
3  | Park Bo-gum       | 22  | Mechanical

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 1
Enter name: Ji Chang-wook
Enter age: 21
Enter course: Civil
```

```
1 student inserted successfully!

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 2
Enter student ID to update: 2
Enter new course: IT
1 student updated successfully!

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 4

ID | Name            | Age | Course
------------------------------------------
1  | Song Joong-ki    | 20  | Computer Science
2  | Kim Seon-ho      | 21  | IT
3  | Park Bo-gum      | 22  | Mechanical
5  | Ji Chang-wook    | 21  | Civil

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 3
Enter student ID to delete: 3
1 student deleted successfully!

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 4

ID | Name            | Age | Course
------------------------------------------
```

```
1  | Song Joong-ki    | 20  | Computer Science
2  | Kim Seon-ho      | 21  | IT
5  | Ji Chang-wook    | 21  | Civil

--- Student Management ---
1. Insert Student
2. Update Student
3. Delete Student
4. View All Students
5. Exit
Enter choice: 5
Exiting...
riddhilahare@Riddhis-MacBook-Air assig8 %
```

Checking in mysql:

```
SELECT * FROM students;
```



```
mysql> select * from students;
+----+---------------+------+------------------+
| id | name          | age  | course           |
+----+---------------+------+------------------+
|  1 | Song Joong-ki |   20 | Computer Science |
|  2 | Kim Seon-ho   |   21 | IT               |
|  5 | Ji Chang-wook |   21 | Civil            |
+----+---------------+------+------------------+
```