

IS230 Tutorial – Project Part II -Report Template JDBC

Student ID: 442106994

Student Name: Sultan Alenzi

Student ID: 442103477

Student Name: Khalid Alherbi

Student ID: 442106013

Student Name: Abdullah Aldakhil (from other section (22142))

1): Code for record **INSERTION**

```
public boolean insert(Student s) {
    // preparing the SQL command
    String cmd = "INSERT INTO student"
    + "(studentID,studentName,department, GPA) "
    + "values("
        + s.getStudentID()
        + ", '"
        + s.getStudentName()
        + "', '" + s.getDeparment()
        + "', " + s.getGPA()
        + ");";

    try {
        // creating statment for execution of command
        Statement stmt = con.createStatement();
        int row = stmt.executeUpdate(cmd);

        // checking the query by getting the rows effected by the command
        if (row > 0)
            System.out.println("Student inserted successfully");
        else {
            // this indicating that some thing wrong happened
            System.out.println("Error while inserted the student row");
            return false;
        }
        stmt.close(); //closing statement
    } catch (SQLException e) {

        // code 1062 indicates that the primary key is already exist
        if (e.getErrorCode() == 1062)
            System.out.println("The primary key exist");
        else {
            // remaining errors
            System.out.println("Error while instered the student row");
            e.printStackTrace();
        }

        return false;
    } // end catch

    // reaching here means every thing is done successfully
    return true;
} // end insert
```

```

public boolean insert(Student s) {
    // preparing the SQL command
    String cmd = "INSERT INTO student"
    + "(studentID,studentName,department, GPA) "
    + "values("
        + s.getStudentID()
        + ", '"
        + s.getStudentName()
        + "', '" + s.getDeparment()
        + "', " + s.getGPA()
        + ");";

    try {
        // creating statment for execution of command
        Statement stmt = con.createStatement();
        int row = stmt.executeUpdate(cmd);

        // checking the query by getting the rows
        effected by the command
        if (row > 0)
            System.out.println("Student inserted
successfully");
        else {
            // this indicating that some thing wrong
            happened
            System.out.println("Error while inserted
the student row");
            return false;
        }
        stmt.close(); //closing statement

    } catch (SQLException e) {

        // code 1062 indicates that the primary key is
        already exist
        if (e.getErrorCode() == 1062)
            System.out.println("The primary key
exist");
        else {
            // remaining errors
            System.out.println("Error while instered
the student row");
            e.printStackTrace();
        }

        return false;
    } // end catch

    // reaching here means every thing is done
    successfully
    return true;
} // end insert

```

2): Code for **DISPLAYING** records

```
public void getAllstudent() {  
  
    Statement stmt;  
    try {  
        stmt = con.createStatement();  
        ResultSet rs = stmt.executeQuery("SELECT * FROM student;");  
  
        //ptinting all records  
        while (rs.next()) {  
            System.out.println();  
            System.out.println("Student Name: " + rs.getString("studentName"));  
            System.out.println("Student ID: " + rs.getInt("studentID"));  
            System.out.println("Department: " + rs.getString("department"));  
            System.out.println("GPA: " + rs.getDouble("GPA"));  
            System.out.println();  
        }  
        stmt.close();  
    } catch (SQLException e) {  
        // TODO Auto-generated catch block  
        e.printStackTrace();  
    }  
}
```

```
public void getAllstudent() {  
  
    Statement stmt;  
    try {  
        stmt = con.createStatement();  
        ResultSet rs = stmt.executeQuery("SELECT *  
FROM student;");  
  
        //ptinting all records  
        while (rs.next()) {  
            System.out.println();  
            System.out.println("Student Name: " +  
rs.getString("studentName"));  
            System.out.println("Student ID: " +  
rs.getInt("studentID"));  
            System.out.println("Department: " +  
rs.getString("department"));  
            System.out.println("GPA: " +  
rs.getDouble("GPA"));  
            System.out.println();  
        }  
        stmt.close();  
  
    } catch (SQLException e) {  
        // TODO Auto-generated catch block  
        e.printStackTrace();  
    }  
}
```

3): Code for **ALTERING** the GPA

- Here to choosing the alter GPA choice

```
case 3:

    System.out.println("Enter the new GPA");
    double newGPA = input.nextDouble();

    if (newGPA > 5 || newGPA < 1) {
        System.out.println("Error GPA");
    } else
        sql.editGPA(newGPA);

    break;
```

case 3:

```
System.out.println("Enter the new GPA");
double newGPA = input.nextDouble();

if (newGPA > 5 || newGPA < 1) {
    System.out.println("Error GPA");
} else
    sql.editGPA(newGPA);

break;
```

- Edit the GPA

```
// this function changes the student GPA
public void editGPA(double newGPA) {
    Statement stmt;

    try {
        stmt = con.createStatement();
        int rowsAffected = stmt.executeUpdate("UPDATE student\r\n"
            + "SET GPA = \r\n" +
            "    CASE \r\n"
            + "        WHEN GPA > " + newGPA + " THEN GPA - 0.10\r\n" + "    "
            + "        WHEN GPA < " + newGPA + " THEN GPA + 0.10\r\n" + "    "
            + "    ELSE GPA\r\n" +
            "    END;");

        if (rowsAffected > 0) { // if rows more than 0 that means the GPA updated successfully
            System.out.println("GPA updated Successfully");
        } else
            System.out.println("GPA not updated");
        stmt.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

```
//this function changes the student GPA
// this function changes the student GPA
public void editGPA(double newGPA) {
    Statement stmt;

    try {
```

```

        stmt = con.createStatement();
        int rowsAffected = stmt.executeUpdate("UPDATE
student\r\n" + "SET GPA = \r\n" + "  CASE \r\n"
        + "      WHEN GPA > " + newGPA + "
THEN GPA - 0.10\r\n" + "      WHEN GPA < " + newGPA
        + " THEN GPA + 0.10\r\n" + "      ELSE
GPA\r\n" + "  END;");

        if (rowsAffected > 0) { // if rows more than 0
that means the the gpa updated successfully
            System.out.println("GPA updated
Successfully");
        } else
            System.out.println("GPA not updated");
        stmt.close();

    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}

```

3): Screenshots of the EXECUTION. Show the menu and examples for the 3 functions.

1- Insertion function:

```
Enter from the following list:
1) insert new student
2) Display All students
3) Alter GPA
4) Exit
Enter your choice: 1
Student - INSERTION
Enter student ID:
3
Enter student Name:
Khalid
Enter student department
IS
Enter student GPA
4.3
Student inserted successfully
```

1.1 The database after insertion

```
MariaDB [studentDb]> select * from student;
+-----+-----+-----+-----+
| studentID | studentName | department | GPA |
+-----+-----+-----+-----+
| 1 | sultan | SWE | 2.00 |
| 2 | fahad | it | 4.30 |
| 3 | khalid | IS | 4.30 |
+-----+-----+-----+-----+
3 rows in set (0.000 sec)
```

1.2 If student entered wrong GPA

```
Enter from the following list:
1) insert new student
2) Display All students
3) Alter GPA
4) Exit
Enter your choice: 1
Student - INSERTION
Enter student ID:
4
Enter student Name:
abeer
Enter student department
SWE
Enter student GPA
9
invalid GPA
```

2- Displaying all students records

```
Enter from the following list:
1) insert new student
2) Display All students
3) Alter GPA
4) Exit
Enter your choice: 2

Student Name: sultan
Student ID: 1
Department: SWE
GPA: 2.0

Student Name: fahad
Student ID: 2
Department: it
GPA: 4.3

Student Name: khalid
Student ID: 3
Department: IS
GPA: 4.3
```

3- Edit GPA

```
Enter from the following list:
1) insert new student
2) Display All students
3) Alter GPA
4) Exit
Enter your choice: 3
Enter the new GPA
3
GPA updated Successfully
```

3.2 before and after updating the GPA

```
Student Name: sultan
Student ID: 1
Department: SWE
GPA: 1.1

Student Name: fahad
Student ID: 2
Department: it
GPA: 4.2

Student Name: khalid
Student ID: 3
Department: IS
GPA: 1.8

Student Name: abdullah
Student ID: 4
Department: hr
GPA: 3.0
```

```
Student Name: sultan
Student ID: 1
Department: SWE
GPA: 1.2

Student Name: fahad
Student ID: 2
Department: it
GPA: 4.1

Student Name: khalid
Student ID: 3
Department: IS
GPA: 1.9

Student Name: abdullah
Student ID: 4
Department: hr
GPA: 3.0
```


4): All the code (الكود كاملا)

```
import java.sql.*;

public class SQL_operations {

    private String user, pwd, url;
    Connection con;

    public SQL_operations() {
        // establishing connection with database
        user = "root";
        pwd = "root";
        url = "jdbc:mariadb://localhost:3306/studentDb";
        try {
            con = DriverManager.getConnection(url, user,
pwd);

        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

    }

    public boolean insert(Student s) {
        // preparing the SQL command
        String cmd = "INSERT INTO student" +
"(studentID,studentName,department, GPA) " + "values(" +
s.getStudentID()
            + ", '" + s.getStudentName() + "', '" +
s.getDeparment() + "', " + s.getGPA() + ");";

        try {
            // creating statment for execution of command
            Statement stmt = con.createStatement();
            int row = stmt.executeUpdate(cmd);

            // checking the query by getting the rows
            effected by the command
            if (row > 0)
                System.out.println("Student inserted
successfully");
            else {
                // this indicating that some thing wrong
                happened
                System.out.println("Error while inserted
the student row");
                return false;
            }
            stmt.close(); // closing statement

        } catch (SQLException e) {
```

```

        // code 1062 indicates that the primary key is
already exist
        if (e.getErrorCode() == 1062)
            System.out.println("The primary key
exist");
        else {
            // remaining errors
            System.out.println("Error while instered
the student row");
            e.printStackTrace();
        }

        return false;
    } // end catch

    // reaching here means every thing is done
successfully
    return true;
} // end insert

public void getAllstudent() {

    Statement stmt;
    try {
        stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT *
FROM student;");

        // ptinting all records
        while (rs.next()) {
            System.out.println();
            System.out.println("Student Name: " +
rs.getString("studentName"));
            System.out.println("Student ID: " +
rs.getInt("studentID"));
            System.out.println("Department: " +
rs.getString("department"));
            System.out.println("GPA: " +
rs.getDouble("GPA"));
            System.out.println();
        }
        stmt.close();

    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

}

public boolean isExist(int sid) {
    Statement stmt;

```

```

        try {
            stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT *
FROM student where studentID=" + sid + ";");

            stmt.close();

            return rs.next(); // this will return true is
student is exist in the set

        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return false;// reaching here means the student is
not exist
    }

    // this function changes the student GPA
    public void editGPA(double newGPA) {
        Statement stmt;

        try {
            stmt = con.createStatement();
            int rowsEffectd = stmt.executeUpdate("UPDATE
student\r\n"
            + "SET GPA = \r\n" +
            "    CASE \r\n"
            + "        WHEN GPA > " + newGPA + " THEN GPA -
0.10\r\n" + "    "
            + "        WHEN GPA < " + newGPA+ " THEN GPA
+ 0.10\r\n" + "    "
            + "    ELSE GPA\r\n" +
            "    END;");

            if (rowsEffectd > 0) { // if rows more that 0
that means the the gpa updated successfully
                System.out.println("GPA updated
Successfully");
            } else
                System.out.println("GPA not updated");
            stmt.close();

        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

    }

    public void close() {

```

```
        try {
            con.close();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

```

import java.util.Scanner;

public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        SQL_operations sql = new SQL_operations();

        Scanner input = new Scanner(System.in);
        int ch = 0;

        while (ch != -1) {
            System.out.println("Enter from the following list:");
            System.out.println("1) insert new student");
            System.out.println("2) Display All students");
            System.out.println("3) Alter GPA");
            System.out.println("4) Exit");

            System.out.print("Enter your choice: ");
            ch = input.nextInt();

            switch (ch) {
                case 1:
                    String sName = "", sDept = "";
                    int sId = 0;
                    double gpa = 0.0;
                    System.out.println("Student - INSERTION");

                    System.out.println("Enter student ID: ");
                    sId = input.nextInt();

                    System.out.println("Enter student Name: ");
                    sName = input.next();

                    System.out.println("Enter student department");
                    sDept = input.next();

                    System.out.println("Enter student GPA");
                    gpa = input.nextDouble();
                    if (gpa > 5 || gpa < 1) {
                        System.out.println("invalid GPA");
                        return;
                    }

                    Student s = new Student(sId, sName, sDept, gpa);
                    sql.insert(s);
                    break;
                case 2:
                    sql.getAllstudent();
                    break;
                case 3:

```

```

        System.out.println("Enter the new GPA");
        double newGPA = input.nextDouble();

        if (newGPA > 5 || newGPA < 1) {
            System.out.println("Error GPA");
        } else
            sql.editGPA(newGPA);

        break;
    case 4:
        sql.close();
        return;
    }
}
}
}
}

```

```

public class Student {
    private int studentID;
    private String studentName , deparment;
    private double GPA;

    public Student(int studentID, String studentName, String
department, double GPA ) {

        //This line to check the GPA value
        if( GPA > 5 || GPA < 1) {
            System.out.println("Enter valid GPA");
            return;
        }else
            this.GPA = GPA;
        this.deparment = department;
    }
}

```

```

        this.studentID = studentID;
        this.studentName = studentName;
    }

    public Student(Student s) {
        studentID = s.studentID;
        studentName = s.studentName;
        deparment = s.deparment;
        GPA = s.GPA;
    }

    public int getStudentID() {
        return studentID;
    }

    public String getStudentName() {
        return studentName;
    }

    public String getDeparment() {
        return deparment;
    }

    public double getGPA() {
        return GPA;
    }

    public void print() {
        System.out.println(

            "Student Name: "+ getStudentName()+
            "\nStudent ID: "+ getStudentID()+
            "\nDepartment: "+getDeparment()+
            "\nGPA: "+getGPA()

        );
    }
}

```