ReadMe

The entry point of the program is Main.java. Main.java connects the database and create all tables if they don't exist. Then it shows the main menu. It also call particular command prompt according to choice which direct to file Menu.java. Menu.java has functions for each management . ClassManagement.java has functions to execute command for Class management. CategoryAndAssignmentManagement.java has functions to execute command for category and assignment management. StudentManagement.java has functions to execute command for student management. And GradeManagement.java has functions to execute command for grade management.

Code snippet:

```java
package com.database;

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

public class GradeManagement {
    public Float showGradesByUsername(Connection conn, Integer classId, String username, Boolean print) {
        try {
            Statement stmt = conn.createStatement();
            String query = String.format( "SELECT c.category_name,c.weight,\n" +
                    "SUM(a.points) AS total_points, \n" +
                    "SUM(s.grade) AS obtained_marks,\n" +
                    "ROUND((SUM(s.grade)/SUM(a.points)),2) as Grade\n" +
                    "FROM Solution s \n" +
                    "JOIN Assignment a \n" +
                    "ON a.id = s.assignment_name \n" +
                    "JOIN Category c \n" +
                    "ON a.category = c.id \n" +
                    "WHERE c.class_id = '%d' AND \n" +
                    "s.username = '%s' \n" +
                    "GROUP BY c.category_name,c.weight ",classId,username);

            ResultSet rs = stmt.executeQuery(query);
            Float total_grade = 0.0F;
            while (rs.next()) {
                if (print){
                    System.out.println("Category: " + rs.getString( "category_name") + " Total points: "+ rs.getInt( "total_points") +
                            " Obtain marks: " + rs.getInt( "obtained_marks") + " Grade: " + rs.getFloat( "Grade"));

                }

                total_grade += rs.getInt( "weight") * rs.getFloat( "Grade");
            }

            if (print){
                System.out.println("Total grade: " + total_grade);
            }

            return total_grade;
```

    1. After you run the program you get a main menu . You can make your choice between 1 to 5.

```
 Enter choice among 1, 2, 3, 4 or 5.

1. Class Management

2. Category and Assignment Management

3. Student Management

4. Grade Reporting

5. Exit
```

2. If you choose 1 , you can give command for class management. If you want to choose other options, you should first activate the class by choosing option 1.

NOTE: After executing the command , you are directed back to main menu. **This is for all commands**

```
Active Class Id: 0
Class Management Command lists
● Create a class: new-class CS410 Sp20 1 "Databases"
● List classes, with the # of students in each: list-classes
● Activate a class:
  o select-class CS410 selects the only section of CS410 in the most recent term, if there is only one such section; if there are multiple sections it fails.
  o select-class CS410 Sp20 selects the only section of CS410 in Fall 2018; if there are multiple such sections, it fails.
  o select-class CS410 Sp20 1 selects a specific section
● show-class shows the currently-active class
select-class CS410
```

3. If you choose 2, You can execute commands to add asignments and categories and show them

show-categories

```
Active Class Id: 1
 Category and Assignment Management Command List
● show-categories – list the categories with their weights
● add-category Name weight – add a new category
● show-assignment – list the assignments with their point values, grouped by category
● add- assignment name Category Description points – add a new assignment
show-categories

All categories
Name: homework Weight: 30
Name: midterm Weight: 15
Name: project Weight: 15
Name: final Weight: 40
```

  o show-assignment

```
 Active Class Id: 6
 Category and Assignment Management Command List
● show-categories - list the categories with their weights
● add-category Name weight - add a new category
● show-assignment - list the assignments with their point values, grouped by category
● add- assignment name Category Description points - add a new assignment
show-assignment
All assignments
Category: final
Name: final Points:100.0
Category: midterm
Name: midterm1 Points:100.0
Name: midterm2 Points:120.0
 Enter choice among 1, 2, 3, 4 or 5.
```

4. If you choose 3, you can execute commands for

- show-students

```
● add-student username — enrolls an already-existing student in the current class. If
the specified student does not exist, report an error.
● show-students - show all students in the current class
● show-students string - show all students with 'string' in their name or username
(case-insensitive)
● grade assignmentname username grade - assign the grade 'grade' for student
with user name 'username' for assignment 'assignmentname'. If the student already has a
grade for that assignment, replace it. If the number of points exceeds the number of
points configured for the assignment, print a warning (showing the number of points
configured)
show-students
 Username: lakhabishal
 Username: sharmaram
```

- show-students string

```
● add-student username studentid Last First — adds a student and enrolls
them in the current class. If the student already exists, enroll them in the class; if the
name provided does not match their stored name, update the name but print a warning
that the name is being changed.
● add-student username — enrolls an already-existing student in the current class. If
the specified student does not exist, report an error.
● show-students - show all students in the current class
● show-students string - show all students with 'string' in their name or username
(case-insensitive)
● grade assignmentname username grade - assign the grade 'grade' for student
with user name 'username' for assignment 'assignmentname'. If the student already has a
grade for that assignment, replace it. If the number of points exceeds the number of
points configured for the assignment, print a warning (showing the number of points
configured)
show-students ram
Student having certain string pattern
 Username: sharmaram first_name: ram
```

- add-student

```
 Active Class Id: 6
Student Management Commands List
● add-student username studentid Last First — adds a student and enrolls
them in the current class. If the student already exists, enroll them in the class; if the
name provided does not match their stored name, update the name but print a warning
that the name is being changed.
● add-student username — enrolls an already-existing student in the current class. If
the specified student does not exist, report an error.
● show-students - show all students in the current class
● show-students string - show all students with 'string' in their name or username
(case-insensitive)
● grade assignmentname username grade - assign the grade 'grade' for student
with user name 'username' for assignment 'assignmentname'. If the student already has a
grade for that assignment, replace it. If the number of points exceeds the number of
points configured for the assignment, print a warning (showing the number of points
configured)
add-student ghimiremadan 1212 ghimire madan


 Student Created Successfully!



 Student Enrolled Successfully!
```

5. If you choose 4, you can look into grade of individual student and Grade for whole class

```
 Active Class Id: 1
● student-grades username - show student's current grade: all assignments, visually
grouped by category, with the student's grade (if they have one). Show subtotals for each
category, along with the overall grade in the class.
● gradebook - show the current class's gradebook: students (username, student ID, and
name), along with their total grades in the class.
student-grades lakhabishal
Category: final Total points: 100 Obtain marks: 95 Grade: 0.95
Category: homework Total points: 150 Obtain marks: 140 Grade: 0.93
Category: midterm Total points: 100 Obtain marks: 92 Grade: 0.92
Category: project Total points: 100 Obtain marks: 95 Grade: 0.95
Total grade: 93.950005
```

```
 Active Class Id: 1
● student-grades username - show student's current grade: all assignments, visually
grouped by category, with the student's grade (if they have one). Show subtotals for each
category, along with the overall grade in the class.
● gradebook - show the current class's gradebook: students (username, student ID, and
name), along with their total grades in the class.
gradebook
Gradebook
Username: lakhabishal Student_id: 1 First name: bishal Last name: lakha Grade: 93.950005
Username: sharmaram Student_id: 1234 First name: ram Last name: sharma Grade: 0.0
```

6. If you choose 5, you will exit from the application

Video URL

https://www.youtube.com/watch?v=fnnYJx9Ucq0