Student Grade Management System

Java Project

Author: Shafiya Munawwar

StudentGradeManagementSystem (Main Class) Source Code

```
//@author Shafiya Munawwar
package studentgrademanagementsystem;
import java.util.Scanner;
//Main class to run the application
public class StudentGradeManagementSystem {
    //Main method to start the program
    public static void main(String[] args) {
        School school = new School(); //Instatntiate the School class to manage students and grade
        Scanner newObject = new Scanner (System.in); //Scanner for user input
        while (true)
        {
            //Infinite loop to display the the main menu
            System.out.println("-----------------------------------------------------");
            System.out.println("Student Grade Management System");
            System.out.println("-----------------------------------------------------");
            System.out.println("1. Student Management");
            System.out.println("2. Grade Management");
            System.out.println("3. Display Information");
            System.out.println("4. Exit");
            System.out.println("-----------------------------------------------------");
            System.out.println("Choose an Option");

            int choice = newObject.nextInt();

            //Handle different menu options using a switch statement
            switch (choice)
            {
                case 1:
                    school.manageStudents(); //Call manageStudents method from the School class
                    break;
                case 2:
                    school.manageGrades(); //Call manageGrades method from the School class
                    break;
                case 3:
                    school.displayInformation(); //Call manageStudents method from the School class
                    break;
                case 4:
                    System.out.println("Exiting....");
                    System.exit(0); //Exit the program
                default:
                    System.out.println("Invalid choice. Please try again."); //Handle invalid choices
            }
        }
```

```
    }

}
```

Student Class Source Code

```
//@author Shafiya Munawwar
package studentgrademanagementsystem;
import java.util.HashMap;
import java.util.Map;
//Class to represent a student
class Student {

    //Define private fields for student name, id, and grades
    private String name;
    private int id;
    private Map<String, Integer> subjectsAndGrades;

    //Constructor to initialize student name and id, and an empty map for subjects and grades
    public Student(String name, int id)
    {
        this.name = name;
        this.id = id;
        this.subjectsAndGrades = new HashMap<>(); //Initialize with an empty map
    }

    //Getter for Name
    public String getName()
    {
        return name;
    }

    //Getter for Id
    public int getId()
    {
        return id;
    }

    //Setter for Name
    public void setName(String name)
    {
        this.name = name;
    }

    //Method to add or update grades for subjects
    public void setGrades(String subject, int grade)
    {
        this.subjectsAndGrades.put(subject, grade);
```

```
    }

    //Method to calculate the average grade
    public double calculateAverageGrade()
    {
        //Check if there are no grades
        if (subjectsAndGrades.isEmpty())
        {
            return 0;
        }
        int sum = 0;
        //Iterate through map values to calculate sum
        for (int grade : subjectsAndGrades.values())
        {
            sum += grade;
        }
        //Calculate average by deviding sum by number of grades
        return (double) sum / subjectsAndGrades.size();
    }

    //Override toString method to provide a string representation of the student
    @Override
    public String toString()
    {
        StringBuilder sb = new StringBuilder();
        sb.append("Student: [ Name = ").append(name).append(" , Id = ").append(id).append(" , Subjects
& Grades = ");
        //Append each subject and grade to the string
        for (Map.Entry < String, Integer > entry : subjectsAndGrades.entrySet()){
            sb.append(entry.getKey()).append(":").append(entry.getValue()).append(", ");
        }
    sb.append("]");
    return sb.toString();
    }
}
```

School Class Source Code

```
//@author Shafiya Munawwar
package studentgrademanagementsystem;
import java.util.ArrayList;
import java.util.Scanner;
//Class to represent a school managing students, and grades
class School {

    //Define a list to store students and a scanner for input
    private ArrayList<Student> students = new ArrayList<>();
    private Scanner newObject = new Scanner (System.in);

    //Method to manage students
    public void manageStudents()
    {
        while (true)
        {
            //Display the student management menu
            System.out.println("-----------------------------------------------------");
            System.out.println("Student Management");
            System.out.println("-----------------------------------------------------");
            System.out.println("1. Add Student");
            System.out.println("2. Update Student");
            System.out.println("3. Remove Student");
            System.out.println("4. View Students");
            System.out.println("5. Back to Main Menu");
            System.out.println("-----------------------------------------------------");
            System.out.println("Choose an Option: ");

            int choice = newObject.nextInt();

            //Handle different menu options using a switch statement
            switch (choice)
            {
                case 1:
                    addStudent(); //Call addStudent method
                    break;
                case 2:
                    updateStudent(); //Call updateStudent method
                    break;
                case 3:
                    removeStudent(); //Call removeStudent method
                    break;
                case 4:
                    viewStudents(); //Call viewStudents method
                    break;
```

```
            case 5:
                return; //Return to the main menu
            default:
                System.out.println("Invalid choice. Please try again."); //Handle invalid choices
        }
    }
}

//Method to add a new student
private void addStudent()
{
    System.out.println("-----------------------------------------------------");
    System.out.println("Enter Student Name: ");
    String name = newObject.next();
    System.out.println("Enter Student ID: ");
    int id = newObject.nextInt();
    //Add new students to the list
    students.add(new Student(name, id));
    System.out.println("Student Added Successfully.");
}

//Method to update an existing student's name
private void updateStudent()
{
    System.out.println("-----------------------------------------------------");
    System.out.println("Enter Student ID to Update: ");
    int id = newObject.nextInt();
    //Iterate through students list to find the student by ID
    for (Student student : students)
    {
        if (student.getId() == id)
        {
            System.out.println("Enter New Name: ");
            String name = newObject.next();
            //Update student name
            student.setName(name);
            System.out.println("Student Updated Successfully.");
            return;
        }
    }
    //Handle case when student is not found
    System.out.println("Student Not Found.");
}

//Method to remove a student
private void removeStudent()
{
```

```
        System.out.println("-----------------------------------------------------");
        System.out.println("Enter Student ID to Remove: ");
        int id = newObject.nextInt();
        //Remove student from the list by ID
        students.removeIf(student -> student.getId() == id);
        System.out.println("Student Removed Successfully.");
    }

    //Method to view all students
    private void viewStudents()
    {
        System.out.println("-----------------------------------------------------");
        System.out.print("");
        //Iterate through students list and print each student
        for (Student student : students)
        {
            System.out.println(student);
        }
    }

    //Method to manage grades
    public void manageGrades()
    {
        System.out.println("-----------------------------------------------------");
        System.out.println("Enter Student ID to Add Grades: ");
        int id = newObject.nextInt();
        //Iterate through students list to find the student by ID
        for (Student student : students)
        {
            if (student.getId() == id)
            {
                System.out.println("Enter Number of Subjects: ");
                int subjects = newObject.nextInt();
                //Iterate to get grades for each subjects
                for (int x = 0; x < subjects; x++)
                {
                    System.out.println("Enter Subject Name: ");
                    String subject = newObject.next();
                    System.out.println("Enter Grade for Subject: ");
                    int grade = newObject.nextInt();
                    //Set grades for the student
                    student.setGrades(subject, grade);
                }
                System.out.println("Grades Added Successfully.");
                return;
            }
        }
```

```
        //Handle case when student is not found
        System.out.println("Student Not Found.");
    }

    //Method to display student information
    public void displayInformation()
    {
        System.out.println("-------------------------------------------------");
        System.out.println("Student Information: ");
        //Iterate through students list and print each student and their average grade
        for (Student student : students)
        {
            System.out.println(student);
            System.out.println("Average Grade: " + student.calculateAverageGrade());
        }
    }
}
```

Output

```
run:
-----------------------------------------------------
Student Grade Management System
-----------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
-----------------------------------------------------
Choose an Option
1
-----------------------------------------------------
Student Management
-----------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
-----------------------------------------------------
Choose an Option:
1
-----------------------------------------------------
Enter Student Name:
Kamala
Enter Student ID:
001
Student Added Successfully.
-----------------------------------------------------
Student Management
-----------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
-----------------------------------------------------
Choose an Option:
1
-----------------------------------------------------
Enter Student Name:
Nimala
Enter Student ID:
002
Student Added Successfully.
```

```
--------------------------------------------------
Student Management
--------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
--------------------------------------------------
Choose an Option:
1
--------------------------------------------------
Enter Student Name:
Vimal
Enter Student ID:
003
Student Added Successfully.
--------------------------------------------------
Student Management
--------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
--------------------------------------------------
Choose an Option:
2
--------------------------------------------------
Enter Student ID to Update:
003
Enter New Name:
Vimala
Student Updated Successfully.
--------------------------------------------------
Student Management
--------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
--------------------------------------------------
Choose an Option:
3
--------------------------------------------------
Enter Student ID to Remove:
```

```
002
Student Removed Successfully.
-----------------------------------------------------
Student Management
-----------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
-----------------------------------------------------
Choose an Option:
4
-----------------------------------------------------
Students: [ Name = Kamala, Id = 1, Subjects & Grades = ]
Students: [ Name = Vimala, Id = 3, Subjects & Grades = ]
-----------------------------------------------------
Student Management
-----------------------------------------------------
1. Add Student
2. Update Student
3. Remove Student
4. View Students
5. Back to Main Menu
-----------------------------------------------------
Choose an Option:
5
-----------------------------------------------------
Student Grade Management System
-----------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
-----------------------------------------------------
Choose an Option
2
-----------------------------------------------------
Enter Student ID to Add Grades:
001
Enter Number of Subjects:
3
Enter Subject Name:
FoP
Enter Grade for Subject:
90
Enter Subject Name:
```

```
SAD
Enter Grade for Subject:
80
Enter Subject Name:
SD
Enter Grade for Subject:
87
Grades Added Successfully.
-----------------------------------------------------
Student Grade Management System
-----------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
-----------------------------------------------------
Choose an Option
2
-----------------------------------------------------
Enter Student ID to Add Grades:
002
Student Not Found.
-----------------------------------------------------
Student Grade Management System
-----------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
-----------------------------------------------------
Choose an Option
2
-----------------------------------------------------
Enter Student ID to Add Grades:
003
Enter Number of Subjects:
3
Enter Subject Name:
FoP
Enter Grade for Subject:
89
Enter Subject Name:
SAD
Enter Grade for Subject:
80
Enter Subject Name:
SD
```

```
Enter Grade for Subject:
70
Grades Added Successfully.
--------------------------------------------------
Student Grade Management System
--------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
--------------------------------------------------
Choose an Option
3
--------------------------------------------------
Student Information:
Students: [ Name = Kamala, Id = 1, Subjects & Grades = SD:87 , FoP:90 , SAD:80 , ]
Average Grade: 85.66666666666667
Students: [ Name = Vimala, Id = 3, Subjects & Grades = SD:70 , FoP:89 , SAD:80 , ]
Average Grade: 79.66666666666667
--------------------------------------------------
Student Grade Management System
--------------------------------------------------
1. Student Management
2. Grade Management
3. Display Information
4. Exit
--------------------------------------------------
Choose an Option
4
Exiting....
BUILD SUCCESSFUL (total time: 2 minutes 44 seconds)
```