
Student Management System Questions

1. Add and Display Students

- Write a program to add new students dynamically to a list, where each student is a Map with fields like `name`, `age`, `rollNumber`, and `grades`. Display all students after adding them.

---

2. Search by Roll Number

- Write a program to search for a student by their `rollNumber`. If the student exists, display their details; otherwise, show a message saying, *"Student not found."*

---

3. Update Grades

- Write a program to update the `grades` of a specific student. For example, if the student has the roll number `101`, update their `maths` grade.

---

4. Calculate Average Grades

- Write a program to calculate the **average grade** of all students for a specific subject, e.g., `maths`.

---

5. Group Students by Grade

- Write a program to group students into categories:
  - `"Excellent"`: Total grade > 300
  - `"Good"`: Total grade between 200 and 300
  - `"Needs Improvement"`: Total grade < 200

---

6. Delete a Student

- Write a program to remove a student from the list using their `rollNumber`. Show the updated list after deletion.

---

7. Add Attendance

- Add an `attendance` key to each student with values like `{'totalDays': 30, 'presentDays': 27}`. Write a program to:
  - Calculate the attendance percentage of each student.
  - Mark students with attendance < 75% as `"Low Attendance"`.

---

8. Generate Report Card

- Write a program to generate a **report card** for a student. Include:
  - Name, roll number, grades for all subjects, and total grade.
  - Attendance percentage and whether the student is **pass** or **fail** based on total grade > 200.

---

9. List All Students in a Grade Range

- Write a program to list all students whose total grades are within a specific range, e.g., 200 to 300.

---

10. Count Students by Age Group

- Count the number of students in different age groups:
  - 15–18 years
  - 19–22 years
  - 23+ years

---

11. Assign Extra-Curricular Activities

- Add an `activities` key to each student, where the value is a list of their extracurricular activities, e.g., `['debate', 'sports']`. Write a program to:
  - Assign activities based on grades (e.g., total grade > 250 gets `"sports"`, otherwise `"art club"`).
  - Display all students with their activities.

---

12. Promote Students

- Write a program to check if students are eligible for promotion based on:
  - Total grade > 300
  - Attendance > 80%

List promoted and non-promoted students.

---

13. Find Top Performers

- Find the top 3 students based on their total grades and display their names, roll numbers, and grades.

---

14. Find Subjects with Lowest Average

- Write a program to calculate the average grade for each subject across all students and find the subject with the lowest average.

---

15. Merge Multiple Classes

- Suppose you have two separate lists for two classes:
- `List<dynamic> classA = [{'name': 'Ali', 'rollNumber': 101, 'grades': {'maths': 85, 'science': 90}}];`
- `List<dynamic> classB = [{'name': 'Ahmed', 'rollNumber': 201, 'grades': {'maths': 80, 'science': 70}}];`

  Write a program to merge both classes into one list without duplicate `rollNumber`.

---

16. Find Missing Data

- Write a program to check if any student is missing required fields like `grades` or `attendance`. Print the names of such students.

---

17. Add Bonus Marks

- Write a program to add bonus marks to all students in a specific subject, e.g., add `5` marks to `science`.

---

18. Partition Students by Pass/Fail

- Write a program to divide students into two lists:
  - Students with total grade > 200 (`Pass`)
  - Students with total grade <= 200 (`Fail`)

---

19. Generate Class Summary

- Write a program to generate a class summary:
  - Total number of students
  - Average grade for the class
  - Percentage of students passing

---

20. Subject-Wise Top Scorer

- Write a program to find the student who scored the highest in each subject.

Example Dataset for Practice

You can use this example dataset for solving the above questions:

```
List<dynamic> students = [
  {'name': 'Ali', 'age': 20, 'rollNumber': 101, 'grades': {'maths': 85,
'science': 90, 'english': 80}, 'attendance': {'totalDays': 30, 'presentDays':
28}},
  {'name': 'Ahmed', 'age': 21, 'rollNumber': 102, 'grades': {'maths': 80,
'science': 70, 'english': 75}, 'attendance': {'totalDays': 30, 'presentDays':
25}},
  {'name': 'Sara', 'age': 22, 'rollNumber': 103, 'grades': {'maths': 90,
'science': 95, 'english': 85}, 'attendance': {'totalDays': 30, 'presentDays':
29}},
];
```

Yeh questions tumhare project ke logical aur analytical aspects ko improve karenge. Agar kisi ka code ya explanation chahiye ho, to batana! 😊