Simplified Project: Student Performance Tracker

Objective:

Create a Java program that tracks multiple students' performance by calculating their total and average marks, and determines if they pass or fail.

Instructions:

1. **Input Details**:
    - Use the `Scanner` class to take input for:
        - Student name.
        - Marks for 3 subjects (integer values between 0 and 100).
2. **Calculate Results**:
    - Compute the **total marks** by summing the 3 subject marks.
    - Compute the **average marks** using type casting if necessary.
3. **Pass/Fail Check**:
    - Define a pass condition: e.g., the student must score at least **40 marks in each subject** and have an **average of 50 or more**.
    - Display "Pass" or "Fail" for the student based on this condition.
4. **Menu with Options**:
    - Option 1: Add a new student and calculate results.
    - Option 2: Display results for all students.
    - Option 3: Exit the program.

Hints for Implementation:

Type Casting:
Use `(double)` to compute the average:

```
double average = (double) total / 3;
```

1.

Pass/Fail Logic:
Use an `if` condition to check both individual subject marks and average:

```
if (subject1 >= 40 && subject2 >= 40 && subject3 >= 40 && average >= 50) {
    System.out.println("Result: Pass");
} else {
    System.out.println("Result: Fail");
}
```

2. **Loop for Menu**:
   Use a `do-while` loop for the menu, and keep the program running until the user chooses to exit.

Expected Output Example:

1. Add New Student
2. Display Results
3. Exit
Enter your choice: 1

Enter student name: Alice
Enter marks for Subject 1: 65
Enter marks for Subject 2: 70
Enter marks for Subject 3: 55

Total Marks: 190
Average Marks: 63.33
Result: Pass

1. Add New Student
2. Display Results
3. Exit

Enter your choice: 2

Student Results:
Name: Alice, Total: 190, Average: 63.33, Result: Pass

1. Add New Student
2. Display Results
3. Exit
Enter your choice: 3

Exiting application...

Why This Works:

- It avoids complexity by replacing grades with a simple pass/fail check.
- Students practice loops, conditionals, and input handling.
- Results are easy to interpret and implement.