

# PYTHON PROJECT



20-11-2024  
ARTIFICIAL INTELLIGENCE

JAWAD SAGHIR  
ZIA KHAN

```
class Student:
    def __init__(self, name, scores):
        self.name = name
        self.scores = scores

    def calculate_average(self):
        if not self.scores:
            return 0
        return sum(self.scores) / len(self.scores)

    def is_passing(self, passing_score=40):
        return all(score >= passing_score for score in self.scores)

class PerformanceTracker:
    def __init__(self):
        self.students = {}

    def add_student(self, name, scores):
        self.students[name] = Student(name, scores)

    def calculate_class_average(self):
        if not self.students:
            return 0
        total_average = sum(student.calculate_average() for student in self.students.values())
        return total_average / len(self.students)

    def display_student_performance(self):
        for student in self.students.values():
            average = student.calculate_average()
            status = "Passing" if student.is_passing() else "Needs Improvement"
            print(f"Student: {student.name}, Average Score: {average:.2f}, Status: {status}")

def main():
    tracker = PerformanceTracker()

    while True:
        name = input("Enter the student's name (or type 'exit' to finish): ")
        if name.lower() == 'exit':
            break

        scores = []
        for subject in ['Math', 'Science', 'English']:
```

```
while True:
    try:
        score = int(input(f"Enter {subject} score for {name}: "))
        if score < 0 or score > 100:
            print("Please enter a score between 0 and 100.")
            continue
        scores.append(score)
        break
    except ValueError:
        print("Invalid input. Please enter a numeric score.")

tracker.add_student(name, scores)

print("\nStudent Performance Report:")
tracker.display_student_performance()

class_average = tracker.calculate_class_average()
print(f"\nClass Average Score: {class_average:.2f}")
```