Venkata Ashita Medam
Mahindra University, Hyderabad

Documentation on Student and Course Management System

This program implements a simple Student and Course Management System using modular object-oriented programming principles.

The provided code adheres to modular object-oriented programming principles by :

- Organizing functionality into separate classes for managing student details and course details.
- Each class encapsulates related attributes and methods, maintaining encapsulation and abstraction.
- Through methods like ‘add_student()’ and ‘remove_student()’, users interact with the system, and the classes abstract away internal implementation details.

About the Code -

Student Class:

- This class represents a student and contains attributes such as name, student ID, branch, year, and grades.
- It includes a method ‘calculate_gpa()’ to calculate the GPA of the student based on their list of grades.

Course Class:

- This class represents a course and contains a list of enrolled students.
- It provides methods to add a student to the course, remove a student from the course, calculate the average GPA of all students in the course, and display the enrolled students.

Usage:

- To create instances of the ‘Student’ class with the required attributes : name, student ID, branch, year, and grades.
- To create instances of the ‘Course’ class with the name of the course.
- To add students to the course using the ‘add_student()’ method.
- To calculate the average GPA of the course using the ‘average_gpa()’ method.

- To display the enrolled students using the ‘show_enrolled_students()’ method.
- To remove a student from the course using the ‘remove_student()’ method.
- Recalculate the average GPA of the course after removal.

Example:

- In the provided example, three students are created and added to the "Machine Learning" course.
- The enrolled students are displayed along with the average GPA of the course.
- One student is removed from the course.
- The enrolled students and the average GPA of the former course is displayed again.

Code Snippets:

```python
# Student Management System
class Student:

  def __init__(self, name, student_id, branch, year, grades):
        self.name = name
        self.student_id = student_id
        self.branch = branch
        self.year = year
        self.grades = grades

  def calculate_gpa(self):
        if len(self.grades) == 0:
            return 0
        total_points = sum(self.grades)
        return total_points / len(self.grades)
```

```python
# Course Management System
class Course:
    def __init__(self, course_name):
        self.course_name = course_name
        self.enrolled_students = []

    def add_student(self, student):
        self.enrolled_students.append(student)

    def remove_student(self, student_id):
        for student in self.enrolled_students[:]:
            if student.student_id == student_id:
                self.enrolled_students.remove(student)
                return True
        return False

    def average_gpa(self):
        if not self.enrolled_students:
            return 0
        total_gpa = sum(student.calculate_gpa() for student in self.enrolled_students)
        return total_gpa / len(self.enrolled_students)

    def show_enrolled_students(self):
        print(f"Students enrolled in {self.course_name}:")
        for student in self.enrolled_students:
            print(f"Name: {student.name}, ID: {student.student_id}")
```

```python
# Example
if __name__ == "__main__":

    # Create students
    student1 = Student("Ashita", 243, "CSE", 3, [8.25, 8.45, 9.20])
    student2 = Student("Sumedha", 222, "CSE", 3, [8.3, 8.6, 9.5])
    student3 = Student("Ananya", 101, "CAM", 3, [7.89, 7.50, 7.7])

    # Create a course
    c1 = Course("Machine Learning")
    c2 = Course("Artificial Intelligence")

    # Adding students
    c1.add_student(student1)
    c1.add_student(student2)
    c1.add_student(student3)
    c1.show_enrolled_students()

    print("\nAverage GPA of the course:", c1.average_gpa())
    print("\n")

    # Removing a student
    c1.remove_student(101)
    c1.show_enrolled_students()

    print("\nAverage GPA of the course after removal:", c1.average_gpa())
```

```
Students enrolled in Machine Learning:
Name: Ashita, ID: 243
Name: Sumedha, ID: 222
Name: Ananya, ID: 101

Average GPA of the course: 8.376666666666665


Students enrolled in Machine Learning:
Name: Ashita, ID: 243
Name: Sumedha, ID: 222

Average GPA of the course after removal: 8.716666666666665
```