Project

On

Student Grade Management System

Project overview

A student grade management system allows teachers to efficiently manage students' grades, assignments, and performance. This Python-based system will take student data, including names, student IDs, and test scores, and calculate the overall grade based on different evaluation criteria.

Using fundamental Python concepts such as variables, loops, functions, and conditional statements, this project helps in developing an understanding of how Python can be used to automate tasks and organize data.

Objectives:-

The primary objectives of this system are:

- Automation: To automate the process of calculating the total score and grade based on input scores for multiple subjects.
- Efficiency: To facilitate the storage and retrieval of student performance data, enabling educators to handle large groups of students with ease.
- Scalability: To design a system that can easily be extended to include additional features such as more subjects, dynamic grade calculation, and exporting data.
- Display the details of all students, including their names, total scores, and grades

Key Python Concepts Used

- Functions: Modularize the code for calculating grades and displaying student details.
- Loops: while loop for adding multiple students, for loop for entering subject scores.
- Conditionals: Used to assign grades based on the total score.
- Lists and Dictionaries: To store student information.

Code Explanation

1. calculate grade Function:-

This function calculates the grade based on the total score. It uses a simple grading system:

- A: Total score ≥ 90
- B: 80 ≤ Total score < 90
- C: 70 ≤ Total score < 80
- D: 60 ≤ Total score < 70
- F: Total score < 60

2. Display students Function:-

- This function prints out the details of each student. It iterates through the list of students and displays their name, student ID, total score, and grade.

Below is the documentation for the **Student Grade Management System** project, explaining the code step-by-step for easy understanding:

Project Documentation: Student Grade Management System

Overview

The **Student Grade Management System** is a Python-based application designed to help teachers manage student data, including names, student IDs, and scores in three subjects. The program calculates the total score and assigns a grade based on predefined grading criteria. It allows teachers to input multiple students' data and displays all student details, including total scores and grades.

Project Requirements

1. **Store student information**:
    - Name
    - Student ID
    - Scores in at least 3 subjects

2. **Implement functionality**:
    - Calculate the total score and overall grade based on predefined criteria.
    - Use loops to allow input for multiple students.
    - Store the data in a dictionary or a list of dictionaries.
    - Display the details of all students, including their names, total scores, and grades.

3. student grade management Function

This is the main function that manages the student data:

- It takes input from the user for the student's name, ID, and scores in 3 subjects.
- It calculates the total score and determines the grade by calling calculate grade.
- Each student's details are stored in a dictionary, and this dictionary is added to a list.
- A loop is used to input multiple students until the user decides to stop.
- Finally, the function calls display students to print all the student details.

4. Running the Program

- The program is started by calling the student grade management() function. It prompts the user to enter the student's information and handles multiple student entries by looping until the user chooses to stop.

Python code:-

```
def calculate_grade(total_score):
    if total_score >= 90:
        return 'A'
    elif total_score >= 80:
        return 'B'
```

```
    elif total_score >= 70:
        return 'C'
    elif total_score >= 60:
        return 'D'
    else:
        return 'F'
def display_students(student_list):
    print("\nStudent Details:")
    for student in student_list:
        print(f"Name: {student['name']}, Student ID: {student['student_id']}, Total Score: {student['total_score']}, Grade: {student['grade']}")
def student_grade_management():
    students = []
    while True:
        name = input("Enter student's name: ")
        student_id = input("Enter student's ID: ")
        scores = []
        for i in range(1, 4):
            score = int(input(f"Enter score for subject {i}: "))
            scores.append(score)
        total_score = sum(scores)
        grade = calculate_grade(total_score)
        student = {
            'name': name,
            'student_id': student_id,
            'total_score': total_score,
            'grade': grade
```

```
        }
        students.append(student)
        more_students = input("Do you want to add another student? (yes/no): ")
        if more_students.lower() != 'yes':
            break

    display_students(students)
student_grade_management()
```

output:-

```
    display_students(students)
student_grade_management()

Enter student's name:  ejjigiri jalandhar
Enter student's ID:  23mca4007
Enter score for subject 1:  98
Enter score for subject 2:  97
Enter score for subject 3:  99
Do you want to add another student? (yes/no):  yes
Enter student's name:  nampally cutie
Enter student's ID:  23mca4099
Enter score for subject 1:  80
Enter score for subject 2:  78
Enter score for subject 3:  77
Do you want to add another student? (yes/no):  yes
Enter student's name:  bekkanti ganesh
Enter student's ID:  23mca4071
Enter score for subject 1:  70
Enter score for subject 2:  66
Enter score for subject 3:  68
Do you want to add another student? (yes/no):  yes
Enter student's name:  domala venkateshwar rao
Enter student's ID:  23mca4008
Enter score for subject 1:  59
Enter score for subject 2:  60
Enter score for subject 3:  58
Do you want to add another student? (yes/no):  no
```

```
Student Details:
Name: ejjigiri jalandhar, Student ID: 23mca4007, Total Score: 294, Grade: A
Name: nampally cutie , Student ID: 23mca4099, Total Score: 235, Grade: A
Name: bekkanti ganesh, Student ID: 23mca4071, Total Score: 204, Grade: A
Name: domala venkateshwar rao, Student ID: 23mca4008, Total Score: 177, Grade: A
```

Conclusion:-

The Student Grade Management System demonstrates the power of Python in solving real-world problems like organizing and managing student data. By automating the grade calculation process, this system reduces the teacher's workload, improves accuracy, and provides a streamlined way to manage multiple students' grades.

The project successfully uses key Python concepts like loops, conditionals, functions, and data structures, making it an excellent example of Python’s utility in data management.