**Exercise 6**

**Aim**

To create a student management system using classes and objects that allows users to enter student details, calculate average scores, and determine results.

**Algorithm**

|  |  |  |
| --- | --- | --- |
| Step 1 | **:** | Start the Program. |
| Step 2 | **:** | Get user input for the number of students. |
| Step 3 | **:** | Create an empty list to store student objects. |
| Step 4 | **:** | Get the name, age, and marks for each student using loops. |
| Step 5 | **:** | Create a student object and add it to the list. |
| Step 6 | **:** | Compute the average score and result using different functions. |
| Step 7 | **:** | Display results for each student. |
| Step 8 | **:** | Stop the Program. |

**Program:**

class Student:

def \_\_init\_\_(self, name, age, marks):

self.name = name

self.age = age

self.marks = marks

def calculate\_average\_score(self):

if len(self.marks) > 0:

return sum(self.marks) / len(self.marks)

else:

return 0

def calculate\_result(self):

average\_score = self.calculate\_average\_score()

if average\_score < 50:

return "Fail"

elif average\_score < 60:

return "Second Class"

elif average\_score < 90:

return "First Class"

else:

return "Distinction"

# Create an empty list to store student objects

students = []

# Collect information for multiple students in a loop

num\_students = int(input("Enter the number of students: "))

for i in range(num\_students):

name = input(f"Enter the name of student {i + 1}: ")

age = int(input(f"Enter the age of student {i + 1}: "))

marks = []

for j in range(2):

subject\_mark = float(input(f"Enter the mark for subject {j + 1} for student {i + 1}: "))

marks.append(subject\_mark)

# Create a Student object and add it to the list

student = Student(name, age, marks)

students.append(student)

# Analyze and display student information, including results

print("\nResult(s):")

for i, student in enumerate(students):

print(f"Student {i + 1}:")

print(f"Name: {student.name}")

print(f"Age: {student.age}")

print(f"Average Score: {student.calculate\_average\_score()}")

result = student.calculate\_result()

print(f"Result: {result}")

**Output**

**Result**

The student management system successfully retrieves and manages student details using classes and objects**.**