

~\Downloads\practice.ipynb - Colab\_files\practicework.py

```
1 #Define a class Student with attributes name, age, score. Create an instance and print its
  attributes.
2 class Student:
3     def __init__(self,name,age,score)
4         self.name=name
5         self.age=age
6         self.score=score
7     student1=("Hassan",15,85)
8     print("Name:",student1.name)
9     print("Age:",student1.age)
10    print("Score:",student1.score)
11 #Add a method display_info() to Student to print a sentence about the student.
12 class Student:
13     def display_info(self):
14         print("we are learning python")
15 #Define a method has_passed() that returns True if score >= 50, else False.
16 def has_passed(score):
17     return score >= 50
18 print(has_passed(75))
19 print(has_passed(40))
20 #Create a second class Teacher with name, subject, and a method teach(). Instantiate and call
  its method.
21 class Teacher:
22     def __init__(self,name,subject)
23         self.name=name
24         self.age=age
25     def teach(self):
26         print(f"{self.name} is teaching {self.subject}.")
27 t1=Teacher("Ms.Mubaraka",Computer)
28 #Modify Student class to include a class variable school = "GUJAR AI School" and print it from
  instances.
29 class student:
30     school = "GUJAR AI School"
31 s=student()
32 print("s.school")
33 #Create two student objects and compare if they are instances of Student using isinstance().
34 class Student:
35     pass
36 s1 = Student()
37 s2 = Student()
38 print(isinstance(s1, Student))
39 print(isinstance(s2, Student))
40 #Create a Course class with attributes title and students (a list). Add a method to add
  students.
41 class Course:
42     def __init__(self, title):
43         self.title = title
44         self.students = []
```

```
45     def add_student(self, student):
46         self.students.append(student)
47 #Write a method in Course to compute the average score of enrolled students.
48 class Course:
49     def __init__(self, title):
50         self.title = title
51         self.students = []
52     def add_student(self, score):
53         self.students.append(score)
54     def average_score(self):
55         return sum(self.students) / len(self.students)
56 #Override the __str__() method in Student class to return a custom string.
57 class Student:
58     def __init__(self, name):
59         self.name = name
60     def __str__(self):
61         return f"Student name is {self.name}"
62 #Use the dir() function to list attributes & methods of an instance of any class.
63 class Student:
64     def __init__(self, name):
65         self.name = name
66 s = Student("Sarmad")
67 print(dir(s))
68
```