FR. CONCEICAO RODRIGUES COLLEGEOFENGINEERIGDepartment of Computer Engineering

Experiment 4 - Python Programs to create class, object and methods 1. Course Details:

Academic	Year 2022 -	Estimated	Time Experiment No. 4 – (
C	Semester S.E. (COMP	Sem. IV Subject	Python Progran
Modul		Chapte	Python Basics
Experiment T	Software Perfor	Subject C	CSL405

Name ofStudent					(2)
Date ofPerforma					
CO	Mapping CSL405.1: Demonstrat basic data structures, functions and (Comprehension)				

Timeline (2)	Preparedness(2)	Effort (2)	Result

2. Aim & Objective of Experiment
To implement following programs in Python.

Objective of experiment 4 is to understand the basic concepts of Class and Objects inPythonProgramming. Students will be able to demonstrate how to create Class and Objects, classandInstance Variables, Class

properties in Python.

Pre-Requisite: Any programming language like C, C++ **Tools:** Python IDE

Python Lab 3 (class, object and methods)

Create a Student class and initialize it with name and roll number. Make methods to : 1. Display - It should display all informations of the student.
2. setAge - It should assign age to student
3. setMarks - It should assign marks to the student.

Code:-

```
class Student:
    def __init__(self,name,roll):
        self.name=name
        self.roll=roll
        self.marks=[]

    def display(self):
        print(f"The info related to student is: {self.name},{self.roll}")
    def setAge(self):
        age=int(input("Enter age:"))
        print(f"Age of {self.name} is {age}")
    def setMarks(self):
        sub=int(input(f"Enter the number of Subjects:"))
        for i in range(sub):
         numbers=int(input(f"Enter the marks in {i+1} subject:"))
         self.marks.append(numbers)
        print(f"Marks of {self.name} is {self.marks}")

o=Student("Gaurav",9612)
o.display()
o.setAge()
o.setMarks()
```

Output:-

```
PS C:\Users\SCI\OneDrive\Desktop\Gaurav Joshi Codes\Py codes> python -u "c:\Users\SCI
oshi Codes\Py codes\classs.py"
The info related to student is: Gaurav,9612
Enter age:19
Age of Gaurav is 19
Enter the number of Subjects:3
Enter the marks in 1 subject:92
Enter the marks in 2 subject:86
Enter the marks in 3 subject:94
Marks of Gaurav is [92, 86, 94]
```

Create a Time class and initialize it with hours and minutes.

1. Make a method addTime which should take two time object and add them. E.g.- (2hour and 50 min)+(1 hr and 20 min) is (4 hr and 10 min)

2. Make a method displayTime which should print the time.
3. Make a method DisplayMinute which should display the total minutes in the Time. E.g.- (1hr 2 min) should display 62 minute.

Code:-

```
class Time:
 def __init__(self,hours,minutes):
    self.hours=int(hours)
    self.minutes=int(minutes)
 def addTime(self,x,y):
    self.tot_hours=(x.hours+y.hours)+(x.minutes+y.minutes)//60
    self.tot_mins=(x.minutes+y.minutes)%60
    print(self.tot_hours,"hours and",self.tot_mins,"minutes.")
 def displayTime(self):
    print(self.hours,"hours and",self.minutes,"minutes.")
 def DisplayMinute(self):
    self.fin_min=0
    for i in range(1,(self.tot_hours)+1):
      self.fin_min=self.fin_min+60
    self.fin_min=self.fin_min+self.tot_mins
    print("Total minutes:",self.fin_min)
t1=Time(2,50)
t2=Time(1,20)
t1.displayTime()
t2.displayTime()
t3=Time(0,0)
t3.addTime(t1,t2)
t3.DisplayMinute()
```

Output:-

```
PS C:\Users\SCI\OneDrive\Desktop\Gaurav Joshi Codes\Py codes> python -
oshi Codes\Py codes\classs.py"
2 hours and 50 minutes.
1 hours and 20 minutes.
4 hours and 10 minutes.
Total minutes: 250
```

Post Lab:

1. Consider the following code:

```
class Clock:
        def __init__(self, time):
                self.time = time
        def print_time(self):
                time = ’6:30’print self.time
clock = Clock(’5:30’)clock.print_time()
```

(a) What does the code print out? If you aren’t sure, create a Python file and run it.

(b) Is that what you expected? Why?

```
The quotation marks used in line 5&6 are wrong also in line 5th self
keyword is not used to access time and also clock.print_time() should be
used in next line
```

2. Consider the following code:

```
class Clock:
        def __init__(self, time):
                self.time = time
        def print_time(self, time):
                print time
clock = Clock(’5:30’)
clock.print_time(’10:30’)
```

(a) What does the code print out? If you aren’t sure, create a Python file and run it.

(b) What does this tell you about giving parameters the same name as object attributes?

```
In the line 5th parentheses and also use of self keyword is missing in
the print statement. In line 6&7 there is improper use of quotation
marks.
```

3. Consider the following code:

```
class Clock:
        def __init__(self, time):
        self.time = time
```

```
        def print_time(self):
        print self.time
boston_clock = Clock(’5:30’)
paris_clock = boston_clock
paris_clock.time = ’10:30’
boston_clock.print_time()
```

(a) What does the code print out? If you aren’t sure, create a Python file and run it.

(b) Why does it print what it does? (Are boston clock and paris clock different objects? Why Or Why not?)

```
In line 3&5 indentation is not proper. In line 5 parentheses are missing. In
line 6 & 8 improper use of Quotation marks.boston_clock and paris_clock are
not different objects as if change is done wrt to any object among the two
the change is reflected to the other object too.
```