# FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)™

**HORMIS NAGAR, MOOKKANNOOR**

**ANGAMALY-683577**

**'FOCUS ON EXCELLENCE'**

# LABORATORY RECORD

## 20MCA131 - PROGRAMMING LAB

**Name:** **ATHIRA RAMACHANDRAN**

**Branch:** **MASTER OF COMPUTER APPLICATIONS**

**Semester:** **1** **Batch: A** **Roll No: 42**

**MARCH 2022**

**FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)™**

**HORMIS NAGAR, MOOKKANNOOR**

**ANGAMALY-683577**

**'FOCUS ON EXCELLENCE'**

# CERTIFICATE

This is to certify that this is the Bonafide record of the Practical work done by Ms. **ATHIRA RAMACHANDRAN** in the **20MCA131 - PROGRAMMING LAB** Laboratory towards the partial fulfilment for the award of the Master Of Computer Applications during the academic year 2021-2022.

Signature of Staff in Charge

Name:

Date:

Signature of H.O.D

Name:

**Date of University practical examination ……**

Signature of

Internal Examiner

Signature of

External Examiner

# CONTENT

| | | | | |
|---|---|---|---|---|
| **12** | | Create a list of colors from comma-separated color names entered by user. Display<br>first and last colors. | **17** | |
| **13** | | Accept an integer n and compute n+nn+nnn. | **18** | |
| **14** | | Print out all colors from color-list1 not contained in color-list2. | **19** | |
| **15** | | Create a single string separated with space from two strings by swapping the character at position 1. | **20** | |
| **16** | | Sort dictionary in ascending and descending order. | **20** | |
| **17** | | Merge two dictionaries. | **21** | |
| **18** | | Find gcd of 2 numbers. | **22** | |
| **19** | | From a list of integers, create a list removing even numbers. | **21** | |
| **20** | | Program to find the factorial of a number | **23** | |
| **21** | | Generate Fibonacci series of N terms | **24** | |
| **22** | | Find the sum of all items in a list | **25** | |
| **23** | | Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square. | **26** | |
| **24** | | Display the given pyramid with step number accepted from user. Eg: N=4<br>1<br>2 4<br>3 6 9<br>4 8 12 16 | **27** | |
| **25** | | Count the number of characters (character frequency) in a string. | **28** | |

| | | | | |
|---|---|---|---|---|
| **26** | | Add 'ing' at the end of a given string. If it already ends with 'ing', then add 'ly' | **30** | |
| **27** | | Accept a list of words and return length of longest word. | **31** | |
| **28** | | Construct following pattern using nested loop<br>*<br>* *<br>* * *<br>* * * *<br>* * * * *<br>* * * *<br>* * *<br>* *<br>* | **32** | |
| **29** | | Generate all factors of a number. | **33** | |
| **30** | | Create a package graphics with modules rectangle, circle and sub-package 3D-graphics with modules cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that finds area and perimeter of figures by different importing statements. (Include selective import of modules and import * statements) | **34** | |
| **31** | | Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area. | **39** | |
| **32** | | Create a Bank account with members account number, name, type of account and balance. Write constructor and methods to deposit at the bank and withdraw an amount from the bank. 3. Create a class Rectangle with private attributes length and width. Overload ' | **41** | |
| **33** | | Create a class Rectangle with private attributes length and width. Overload ' | **43** | |
| **34** | | Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time. | **45** | |

| | | | | |
|---|---|---|---|---|
| **35** | | Create a class Publisher (name). Derive class Book from Publisher with attributes title and author. Derive class Python from Book with attributes price and no_of_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overriding. | **46** | |
| **36** | | Write a Python program to read a file line by line and store it into a list. | **48** | |
| **37** | | Write a Python program to read specific columns of a given CSV file and print the content of the columns. | **49** | |

## C01

**Experiment Number:1**

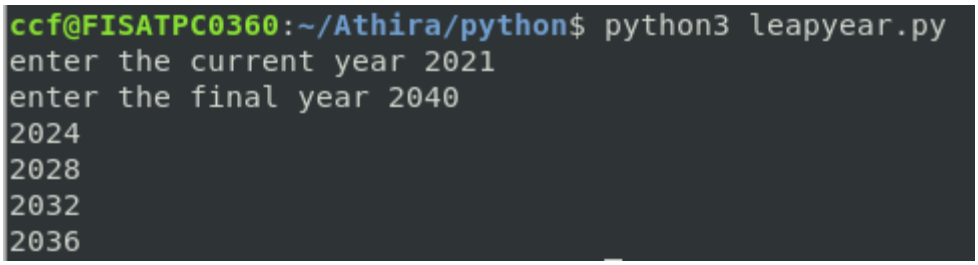
**Aim:**

1) Display future leap years from current year to a final year entered by user

**Program Code:**

```
currentyear=int(input("enter the current year"))
finalyear=int(input("enter the final year"))
for year in range(currentyear,finalyear):
        if(year%400==0)or(year%100!=0)and(year%4==0):
                print(year)
```

**output:**




**Experiment Number:2**

**Aim:**

2)list comprehensions:

a)Generate positive list of numbers from a given list of integers

b)Square of N numbers

c)Form a list of vowels selected from a given period

d)List ordinal value of each element of a word

**Program code:**

```
print("elements in the list are")
list1=[11,-24,0,27,-30]

for num in list1:

        if num>=0:
                print(num)



print("elements in the list are")
list1=[3,6,9,12,15]

for s in list1:

        num=s*s
        print(num)



l=[]
word=input("enter a word")

for i in word:

        if i in "aAeEiIOoUu":

                l.append(i)

print(l)



word=input("enter a word")

for i in word:

        print(ord(i))
```

**output:**

```
ccf@FISATPC0360:~/Athira/python$ python3 co13.py
elements in the list are
11
0
27
elements in the list are
9
36
81
144
225
enter a word welcome
['e', 'o', 'e']
enter a word great
32
103
114
101
97
116
ccf@FISATPC0360:~/Athira/python$
```

**Experiment Number:3**

**Aim:**

3)Count the occurences of each word in a line of text.

**Program Code:**

```
list1=[]
list2=[]
x=input("Enter a string:")
for i in x.split(" "):
  list1.append(i)
  if i not in list2:
    list2.append(i)
for i in list2:
  print(i,"\t",list1.count(i))
```

**output:**

```
user@user-desktop:~/Athira42$ python3 occurrence.py
Enter a string:india is my country india is
india    2
is       2
my       1
country          1
```

**Experiment Number:4**

**Aim:**

4)prompt the user for a list of integers.For all values greater than 100,store ‘over’ instead

**Program Code:**

```
list=[]

print("enter 4 integer numbers")

for i in range(4):
        j=int(input())
        if j>100:

                list.append('over')
        else:

                list.append(j)

print (list)
```

**output:**

```
ccf@FISATPC0360:~/Athira/python$ python3 over.py
enter 4 integer numbers
101
45
200
68
['over', 45, 'over', 68]
```

**Experiment Number:5**

**Aim:**

5)Store a list of first names.count the occurences of ‘a’ within the list

**Program Code:**

```
list=[]
l=[]
print("enter 5 names")
for i in range(5):

        list.append(input())

for i in list:

        count=0
        for j in i:

                if(j=='a'):

                        count=count+1

        l.append(count)

print(l)
```

**output**:

```
ccf@FISATPC0360:~/Athira/python$ python3 occurence.py
enter 5 names
athira
arya
sreeja
anagha
feji
[2, 2, 1, 3, 0]
```

**Experiment Number:6**

**Aim:**

6)Enter 2 list of integers

a)whether list are of same length

b)whether list sums to same value

c)whether any value occur in both

**Program code:**

```
l1=[2,6,3,9,]
l2=[4,5,6]
x=len(l1)
y=len(l2)
```

```
if x==y:
        print("list are of same length")

else:
        print("list are of different length")

s1=0
s2=0
for i in range(x):
        s1=s1+l1[i]
print("the sum of 1st list:",s1)
for j in range(y):
        s2=s2+l2[j]
print("The sum of second list:",s2)

if s1==s2:
        print("sum of list are same")
else:
        print("sum of list are different")

for i in range (x):
        for j in range (y):
                if l1[i]==l2[j]:
                        print(l1[i], "and" ,l2[j],"occur in both")
```

**output**:

```
stud@debian:~/ATHIRA42/python$ python3 program7.py
list are of different length
the sum of 1st list: 20
The sum of second list: 15
sum of list are different
6 and 6 occur in both
```

**Experiment Number:7**

**Aim:**

7)Get a string from an input string where all occurences of first character replaced with '$',except first character.

**Program Code:**

```
str1=input("enter a string ")
print("original string:",str1)
char=str1[0]
str1=str1.replace(char,'$')
str1=char+str1[1:]
print("Replaced string:",str1)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 onion.py
enter a string onion
original string: onion
Replaced string: oni$n
```

**Experiment Number:8**

**Aim:**

8)create a string from given string where first and last characters exchanged.

**Program Code:**

```
s="python"
t=s[0]
t1=s[-1]
n=len(s)
ns=t1+s[1:n-1]+t
print(ns)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 python.py
nythop
```

**Experiment Number:9**

**Aim:**

9) Accept the radius from user and find area of circle

**Program Code:**

```
p=int(input("enter the radius- "))

a=3.14*p*p

print("Area of Circle= ",a)
```

**output:**

```
stud@debian:~/ATHIRA$ python3 area.py
enter the radius- 5
Area of Circle=  78.5
```

**Experiment Number:10**

**Aim:**

10) Find biggest of 3 numbers entered

**Program Code:**

```
a=int(input("enter the first number"))
b=int(input("enter the second number"))
c=int(input("enter the third number"))
if a>b:
        if a>c:
                print(a)
        else:
                print(c)
else:
        if b>c:
                print(b)
        else:
                print(c)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 biggest3.py
enter the first number7
enter the second number9
enter the third number5
9
```

**Experiment Number:11**

**Aim:**

11)Accept a file name from user and print extension of that.

**Program Code:**

```
import os
a=input("Enter the file name:")
print("The extension of file",a,"is",os.path.splitext(a))
```

**output:**

```
user@user-desktop:~/Athira42$ python3 extension.py
Enter the file name:swap.py
The extension of file swap.py is ('swap', '.py')
```

**Experiment Number:12**

**Aim:**

12)create a list of colors from cpmma separated color names entered by user. Display first and last colors

**Program Code:**

```
color=[]
color=[i for i in input("enter the color:").split(',')]
print(color)
i=len(color)-1
print("First color:",color[0])
print("last color:",color[i])
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 program13.py
enter the color:red,green,yellow,blue,pink
['red', 'green', 'yellow', 'blue', 'pink']
First color: red
last color: pink
```

**Experiment Number:13**

**Aim:**

13)Accept an integer n and compute n+nn+nnn

**Program Code:**

```
x=int(input("Enter an integer"))

n1=str(x)
n2=n1+n1
n3=n2+n1

result=int(n1)+int(n2)+int(n3)
print(result)
```

**output:**

```
user@user-desktop:~/Athira42$ python3 14.py
Enter an integer:4
492
```

**Experiment Number:14**

**Aim:**

14)print out of all colors from color-list1 not contained in color-list2

**Program Code:**

```
l1=['pink','red','black','green','blue']
l2=['pink','yellow','red','violet']
l3=[]

flag=0
print('colours not in l2 are')

for i in l1:

        if i not in l2:

                l3.append(i)

                flag=1

print(l3)

if flag==0:

        print('not present')
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 colorlist.py
colours not in l2 are
['black', 'green', 'blue']
```

**Experiment Number:15**

**Aim:**

15)Create a single separated with space from two strings by swapping the character at position 1.

**Program Code:**

```
str1=input("Enter first string:")
str2=input("Enter second string:")
str3=str2[0]+str1[1:]+" "+str1[0]+str2[1:]
print(str3)
```

**output:**

```
user@user-desktop:~/Athira42$ python3 swap.py
Enter first string:Athira
Enter second string:Ramachandran
Rthira Aamachandran
```

**Experiment Number:16**

**Aim:**

16)Sort dictionary in ascending and descending order.

**Program Code:**

```
dict1={"a":1,"c":3,"d":2,"b":4}
l=list(dict1.items())
print(l)
l.sort()
print("Ascending order is\n",l)
l=list(dict1.items())
l.sort(reverse=True)
print("Descending order is\n",l)
```

**output:**

```
user@user-desktop:~/Athira42$ python3 sortdic.py
[('a', 1), ('c', 3), ('d', 2), ('b', 4)]
Ascending order is
 [('a', 1), ('b', 4), ('c', 3), ('d', 2)]
Descending order is
 [('d', 2), ('c', 3), ('b', 4), ('a', 1)]
```

**Experiment Number:17**

**Aim:**

17)Merge two dictionaries.

**Program Code:**

```
dict1={"Name":"Anu","Age":40}

dict2={"Gender":"F","Qualification":"PG"}

dict1.update(dict2)

print(dict1)
```

**output:**

```
user@user-desktop:~/Athira42$ python3 mergedic.py
{'Name': 'Anu', 'Age': 40, 'Gender': 'F', 'Qualification': 'PG'}
```

**Experiment Number:18**

**Aim:**

18)Find gcd of 2 numbers

**Program Code:**

```
x=int(input("enter the first number"))
y=int(input("enter the second number"))
if x>y:
        smallest=y
else:
        smallest=x
for i in range(1,smallest+1):
        if((x%i==0)and(y%i==0)):
                hcf=i
print(hcf)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 gcd.py
enter the first number12
enter the second number4
4
```

**Experiment Number:19**

**Aim:**

19)From a list of integers,create a list removing even numbers.

**Program Code:**

```
l1=[1,2,3,4,5,6]
l2=[]
for i in l1:
  if i%2!=0:
    l2.append(i)
print(l2)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 evenremove.py
[1, 3, 5]
```

**C02**

**Experiment Number:20**

**Aim:**

20)Program to find the factorial of a number.

**Program Code:**

```
n=int(input("enter the number"))
fact=1
for i in range(1,n+1):
        fact=fact*i
print(fact)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 factorial.py
enter the number4
24
```

**Experiment Number:21**

**Aim:**

21)Generate Fibonacci series of N terms.

**Program Code:**

```
n=int(input("enter the number"))
f1=0
f2=1

print(f1)

print(f2)

for i in range(0,n):

        f3=f1+f2
        print(f3)
        f1=f2
        f2=f3
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 fibseries.py
enter the number8
0
1
1
2
3
5
8
13
21
34
```

**Experiment Number:22**

**Aim:**

22)Find the sum of all items in a list.

**Program Code:**

```
list1=[1,2,3,4,5]
sum=0
for i in list1:
  sum=sum+i
print(sum)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 sumlist.py
15
```

**Experiment Number:23**

**Aim:**

23)Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.

**Program Code:**

```
limit1=1000
limit2=9999
list1=[]
for i in range(limit1,limit2):
        j=i
        digit=[]
        while(i!=0):
                digit.append(i%10)
                i=int(i/10)
        count=0
        for n in digit:
                if n%2==0:
                        count=count+1
```

```
if count==4:
                for k in range(31,100):
                        if((k**2)==j):
                                list1.append(j)
                                print(k)
print(list1)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 co2p4.py
68
78
80
92
[4624, 6084, 6400, 8464]
```

**Experiment Number:24**

**Aim:**

24)Display the given pyramid with step number accepted from user.

EG:N=4

1

2 4

3 6 9

4 8 12 16

**Program code:**

```
n=int(input("Enter a number:"))
for i in range(1,n+1):
  for j in range(i,(i*i)+1,i):
    print(j,"\t",end="")
  print("\n")
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 stepnumber.py
Enter a number:5
1

2       4

3       6       9

4       8       12      16

5       10      15      20      25
```

**Experiment Number:25**

**Aim:**

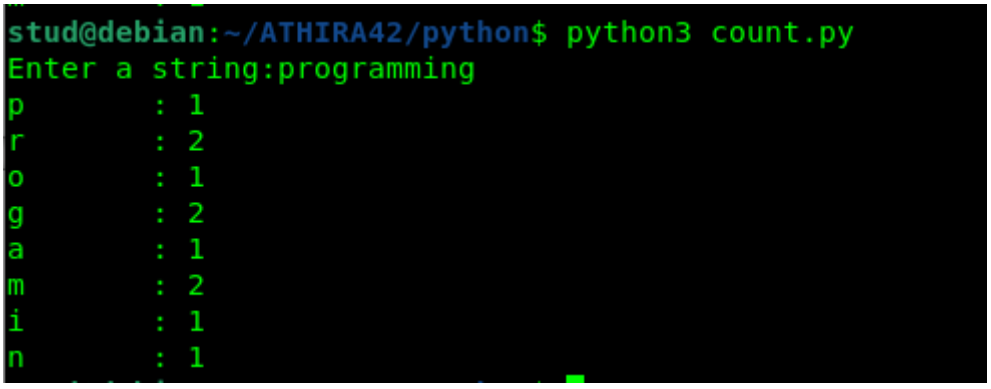
25)Count the number of characters (character frequency)in a string.

**Program Code:**

```
string=input("Enter a string:")
list1=[]
for i in string:
  if i not in list1:
    list1.append(i)

for i in list1:
  count=0
  for j in string:
    if(i==j):
      count=count+1
  print(i,"\t:",count)
```

**output:**



**Experiment Number:26**

**Aim:**

26)Add ‘ing’ at the end of a given string.If it already ends with ‘ing’,then add ‘ly’

**Program code:**

```
string=input("Enter a string:")
if(string[-3:]=="ing"):
  string+="ly"
else:
  string+="ing"
print(string)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 ing.py
Enter a string:coming
comingly
stud@debian:~/ATHIRA42/python$ python3 ing.py
Enter a string:welcome
welcomeing
```

**Experiment Number:27**

**Aim:**

27)Accept a list of words and return length of longest word.

**Program Code:**

```
lis=[]
n=int(input("Enter the range:"))

print("Enter the words:")

for i in range(0,n):

  lis.append(input(""))

longest=lis[0]
for i in range(1,n):

  if(len(lis[i])>len(longest)):

    longest=lis[i]

print("Length of longest word is",len(longest))
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 longest.py
Enter the range:4
Enter the words:
athira
anu
akshaya
diya
Length of longest word is 7
```

**Experiment Number:28**

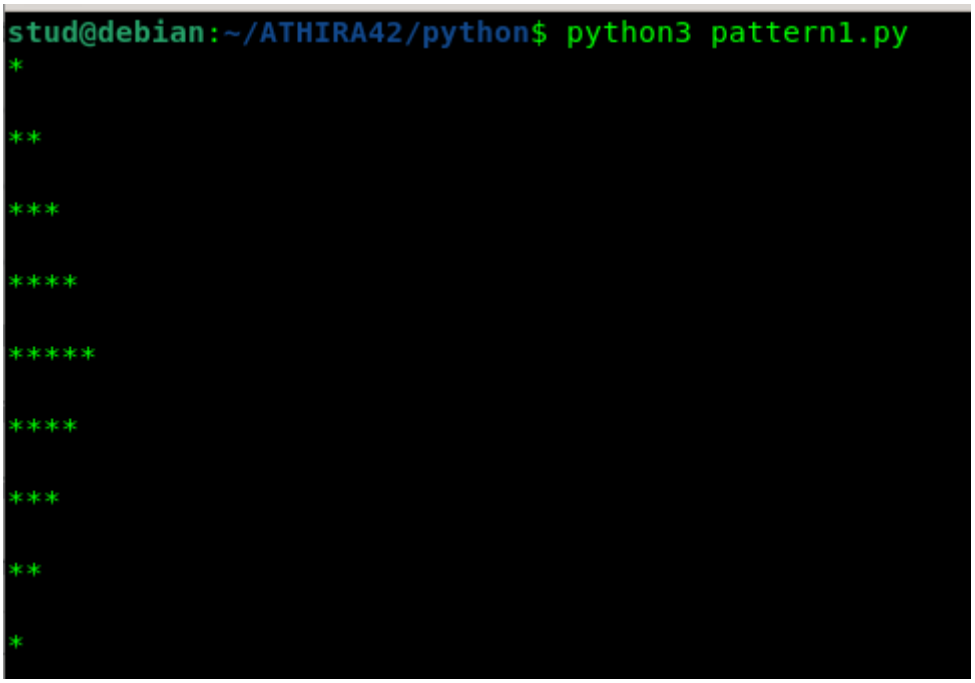
**Aim:**

28)Construct following pattern using nested loop.

**Code:**

```
k='*'
for i in range(1,6):
        for j in range(1,i+1):
                print(k,end="")
        print("\n")
for i in range(5,0,-1):
        for j in range(1,i):
                print(k,end="")
        print("\n")
```

**output:**




**Experiment Number:29**

**Aim:**

29)Generate all factors of a number.

**Program Code:**

```
n=int(input("Enter a number:"))
print("Factors are")
for i in range(1,n+1):
  if(n%i==0):
```

```
print(i)
```

**output:**

```
stud@debian:~/ATHIRA42/python$ python3 factor.py
Enter a number:6
Factors are
1
2
3
6
```

**C03**

**Experiment Number:30**

**Aim:**

30)Create a package graphics with modules rectangle, circle and sub-package 3D-graphics

with modules cuboid and sphere. Include methods to find area and perimeter of respective

figures in each module. Write programs that finds area and perimeter of figures by different

importing statements. (Include selective import of modules and import * statements)

**Program Code:**

**Graphics\circle.py**

```
from math import pi

def area_circle(radius):

return pi*radius*radius


def perimeter_circle(radius):
return 2*pi*radius
```

**Graphics\rectangle.py**

```
def area_rec(length,width):
return length*width



def perimeter_rec(length,width):
return 2*(length+width)
```

**Graphics\tdgraphics\cuboid.py**

```
def area_cuboid(l,b,h):

return 2*(l*h + b*h + l*b)

def volume_cuboid(l,b,h):

return l*b*h
```

**Graphics\tdgraphics\sphere.py**

```
from math import pi

def area_sphere(radius):

return 4*(pi*radius*radius)

def perimeter_sphere(radius):

return 2*pi*radius
```

**graphics.py (driver code)**

```
import Graphics
from Graphics import circle,rectangle
from Graphics.tdgraphics import cuboid,sphere
from Graphics.circle import *


print("Area of a Rectangle with length and width 10 is : ",rectangle.area_rec(10,10))
print("Perimeter of a Rectangle with length and width 10 is :
",rectangle.perimeter_rec(10,10))
print("\n")



print("Area of a circle with radius 10 is : ",circle.area_circle(10))
print("Perimeter of a circle with radius 10 is ",circle.perimeter_circle(10))
print("\n")



print("Area of a cuboid with length,width,height 10 is :
",cuboid.area_cuboid(10,10,10))
print("Volume of a cuboid with length,width,height 10 is :
",cuboid.volume_cuboid(10,10,10))
print("\n")



print("Area of a sphere with radius 10 is : ",sphere.area_sphere(10))
print("Perimeter of a sphere with radius 10 is ",sphere.perimeter_sphere(10))
```

**output:**

```
C:\Users\ATHIRA RAMACHANDRAN>md Graphics

C:\Users\ATHIRA RAMACHANDRAN>cd Graphics

C:\Users\ATHIRA RAMACHANDRAN\Graphics>notepad __init__.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics>notepad rectangle.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics>notepad circle.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics>md tdgraphics

C:\Users\ATHIRA RAMACHANDRAN\Graphics>cd tdgraphics

C:\Users\ATHIRA RAMACHANDRAN\Graphics\tdgraphics>notepad __init__.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics\tdgraphics>notepad cuboid.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics\tdgraphics>notepad sphere.py

C:\Users\ATHIRA RAMACHANDRAN\Graphics\tdgraphics>cd . .

C:\Users\ATHIRA RAMACHANDRAN\Graphics\tdgraphics>cd . .
```

```
C:\Users\ATHIRA RAMACHANDRAN>python graphics.py
Area of a Rectangle with length and width 10 is :  100
Perimeter of a Rectangle with length and width 10 is :  40


Area of a circle with radius 10 is :  314.1592653589793
Perimeter of a circle with radius 10 is  62.83185307179586


Area of a cuboid with length,width,height 10 is :  600
Volume of a cuboid with length,width,height 10 is :  1000


Area of a sphere with radius 10 is :  1256.6370614359173
Perimeter of a sphere with radius 10 is  62.83185307179586
```

## C04

**Experiment Number:31**

**Aim:**

31)Create Rectangle class with attributes length and breadth and methods to find area and perimeter.Compare two rectangle objects by their area.

**Program Code:**

```
class rectangle:
        def __init__(self,length,breadth):
                self.length=length
                self.breadth=breadth
        def area(self):
                return self.length*self.breadth
        def perimeter(self):
                return 2*(self.length+self.breadth)

r1=rectangle(6,4)

r2=rectangle(10,7)

x=r1.area()

y=r2.area()

z=r1.perimeter()
```

```
w=r2.perimeter()

print("Area of rectangle1 is",x)

print("Area of rectangle2 is",y)

print("Perimeter of rectangle1 is",z)

print("Perimeter of rectangle2 is",w)

if(x>y):

        print("Area of rectangle 1 is larger")
else:

        print("Area of rectangle 2 is larger")
```

**Output:**

```
Area of rectangle1 is 24
Area of rectangle2 is 70
Perimeter of rectangle1 is 20
Perimeter of rectangle2 is 34
Area of rectangle 2 is larger
```

**Experiment Number:32**

**Aim:**

32)Create a Bank account with members account number,name,type of account and balance.Write constructor and methods to deposit at the bank and withdraw an amount from the bank.

**Program Code:**

```
class bank:

    def __init__(self,acc_no,name,acc_type,balance):
        self.acc_no=acc_no
        self.name=name
        self.type=acc_type
        self.balance=balance
    def withdrawal(self,x):
        self.balance=self.balance-x
        print("Balance amount after withdrawal:",self.balance)

    def deposit(self,y):

        self.balance=self.balance+y
        print("Balance amount after deposit:",self.balance)

    def display(self):

        print("Account Number:",self.acc_no)
        print("Account Name:",self.name)
        print("Account Type:",self.type)
        print("Account Balance:",self.balance)
```

```
account1=bank(1234,"Anu","Savings",25000)
account2=bank(3456,"Ammu","Savings",5000)

account3=bank(7890,"Anju","Savings",15000)

account4=bank(4587,"Athira","Savings",4000)

account1.deposit(10000)

account1.withdrawal(2000)

account2.deposit(5000)

account2.withdrawal(1000)

account3.deposit(15000)

account3.withdrawal(3000)

account4.deposit(20000)

account4.withdrawal(10000)
```

**Output:**

```
Balance amount after deposit: 35000
Balance amount after withdrawal: 33000
Balance amount after deposit: 10000
Balance amount after withdrawal: 9000
Balance amount after deposit: 30000
Balance amount after withdrawal: 27000
Balance amount after deposit: 24000
Balance amount after withdrawal: 14000
```

**Experiment Number:33**

**Aim:**

33)Create a class Rectangle with private attributes length and width.Overload '<' operator to compare the area of 2 rectangles.

**Program Code:**

```
class rectangle:
        def __init__(self,length,breadth):
                self.__length=length
                self.__breadth=breadth
        def area(self):
                return self.__length*self.__breadth
        def perimeter(self):
                return 2*(self.__length+self.__breadth)
        def __lt__(self,r2):
                if(self.__length*self.__breadth<r2.__length*r2.__breadth):
                        return True
                else:
                        return False

r1=rectangle(10,6)
r2=rectangle(6,4)
x=r1.area()
y=r2.area()
z=r1.perimeter()
```

```
w=r2.perimeter()

print("Area of rectangle 1 is",x)

print("Area of rectangle 2 is",y)

print("Perimeter of rectangle 1 is",z)



print("Perimeter of rectangle 2 is",w)

if(r1<r2):

        print("Rectangle 1 is smaller")
else:

        print("Rectangle 2 is smaller")
```

**Output:**

```
Area of rectangle 1 is 60
Area of rectangle 2 is 24
Perimeter of rectangle 1 is 32
Perimeter of rectangle 2 is 20
Rectangle 2 is smaller
```

**Experiment Number:34**

**Aim:**

34)Create a class Time with private attributes hour,minute and second.Overload '+' operator to find sum of 2 time.

**Program Code:**

```
class Time:

    def __init__(self,hour,minute,second):
        self.__hour=hour
        self.__minute=minute
        self.__second=second
    def __add__(self,t2):
        a=self.__hour=t1.__hour+t2.__hour
        b=self.__minute=t1.__minute+t2.__minute
        c=self.__second=t1.__second+t2.__second
        print("The Sum of Two Times is",a,b,c)

t1=Time(1,30,13)
t2=Time(4,10,25)
t3=t1+t2
```

**Output:**

```
The Sum of Two Times is 5 40 38
```

**Experiment Number:35**

**Aim:**

35)Create a class publisher(name).Derive class Book from publisher with attributes title and author.Derive class python from book with attributes price and no of pages. Write a program that displays information about a python book.Use base class constructor invocation and method overriding.

**Program Code:**

```
class Publisher(object):
    def __init__(self,name):
        self.name=name
    def display1(self):
        print(self.title)
        print(self.author)

class Book(Publisher):
    def __init__(self,name,title,author):
        super().__init__(name)
        self.title=title
        self.author=author
    def display2(self):

        super().display1()

        print(self.title)

        print(self.author)
```

```
class Python(Book):
    def __init__(self,name,title,author,price,no_of_pages):
        super().__init__(name,title,author)
        self.price=price
        self.no_of_pages=no_of_pages
    def display3(self):
        super().display2()
        print(self.price)
        print(self.no_of_pages)


  p=Python("ABC Publications","Taming Python","jeeva jose",100,500)
p.display3()
#p.display2()
#p.display3()
```

**Output:**

```
Taming Python
jeeva jose
Taming Python
jeeva jose
100
500
```

**CO5**

**Experiment Number:36**

**Aim:**

36)Write a Python program to read a file line by line and store it into a list.

**Program Code:**

```
f=open("text_file.txt",'r')
lines=[]
for line in f:
        lines.append(line.strip())
print(lines)
```

**output:**

```
C:\Users\ATHIRA RAMACHANDRAN\OneDrive\Documents\PYTHON>python text.py
['Education is the process of facilitating learning, or the acquisition of knowledge, skills, values, morals, beliefs, h
abits, and personal development. Educational methods include teaching, training, storytelling, discussion, and directed
research. ... In most regions, education is compulsory up to a certain age.']
```

**Experiment Number:37**

**Aim:**

37)Write a Python program to read each row from a given csv file and print a list of strings.

**Program Code:**

```
import csv

with open('data.csv','r')as file:

    reader=csv.reader(file)

    for row in reader:
        print(row)
```

**output:**

```
C:\Users\ATHIRA RAMACHANDRAN\OneDrive\Documents\PYTHON>python second.py
['NAME', 'AGE', 'QUALIFICATION']
['Anu', '21', 'PG']
['Sandra', '19', 'DEGREE']
['Anandhu', '18', 'DIPLOMA']
['Swathy', '21', 'BTECH']
['Anna', '19', 'DIPLOMA']
```