Practical File

Problem-Solving using Python Programming

Sub Code: 24CSE0101

Submitted

in the partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINNERING

CHITKARA UNIVERSITY

CHANDIGARH-PATIALA NATIONAL HIGHWAY RAJPURA (PATIALA) PUNJAB-140401 (INDIA)

November, 2024

Submitted To:

Faculty Name: **Dr.Susama Bagchi**

Designation: **Professor (CSE)**

Chitkara University, Punjab

Submitted By:

Name: **Ritesh Jaswal**

Roll.No**.: 2410991816 G-23**

Sem-1 Batch 2024

INDEX

Sr.no	Practical Name	Teacher Sign
1.	a) Write a Python Program to Calculate the Area of a Triangle b) Write a Python Program to Swap Two Variables c) Write a Python Program to Convert Celsius to Fahrenheit	
2.	a.) Write a Python Program to Check if a Number is Odd or Even b.) Write a Python Program to Check if a Number is Positive, Negative or 0 c.) Write a Python Program to Check Armstrong Number	
3.	a.) Write a Python program to check if a given number is Fibonacci number? b.) Write a Python program to print cube sum of first n natural numbers. c.) Write a Python program to print all odd numbers in a range.	
4.	a.) Write a Python Program to Print Pascal Triangle b.) WAP to Draw the following Pattern for n number: 1 1 1 1 1 2 2 2 2 3 3 3 4 4 5	
5.	Write a program with a function that accepts a string from keyboard and create a new string after converting character of each word capitalized.	
6.	**a.)** Write a program that accepts a list from user. Your program should reverse the content of list and display it. Do not use reverse () method. **b)** Find and display the largest number of a list without using built-in function max (). Your program should ask the user to input values in list from keyboard.	
7.	Find the sum of each row of matrix of size m x n.	
8.	**a)** Write a program that reads a string from keyboard and display: * The number of uppercase letters in the string. * The number of lowercase letters in the string. * The number of digits in the string. * The number of whitespace characters in the string. **b)** Python Program to Find Common Characters in Two Strings. **c)** Python Program to Count the Number of Vowels in a String	

9.	a) Write a Python program to check if a specified element presents in a tuple of tuples. Original list: ((‘Red’ ,’White’ , ‘Blue’),(‘Green’, ’Pink’ , ‘Purple’), (‘Orange’, ‘Yellow’, ‘Lime’)) Check if White present in said tuple of tuples! True Check if Olive present in said tuple of tuples! False b) Write a Python program to remove an empty tuple(s) from a list of tuples. Sample data: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')] Expected output: [('',), ('a', 'b'), ('a', 'b', 'c'), 'd']	
10.	a) Write a Program in Python to Find the Differences Between Two Lists Using Sets.	
11.	a) Write a Python program Remove duplicate values across Dictionary Values. Input : test_dict = {‘Manjeet’: [1], ‘Akash’: [1, 8, 9]} Output : {‘Manjeet’: [], ‘Akash’: [8, 9]} Input : test_dict = {‘Manjeet’: [1, 1, 1], ‘Akash’: [1, 1, 1]} Output : {‘Manjeet’: [], ‘Akash’: []} b) Write a Python program to Count the frequencies in a list using dictionary in Python. Input : [1, 1, 1, 5, 5, 3, 1, 3, 3, 1,4, 4, 4, 2, 2, 2, 2] Output : 1 : 5 2 : 4 3 : 3 4 : 3 5 : 2 Explanation : Here 1 occurs 5 times, 2 occurs 4 times and so on...	
12.	a) Write a Python Program to Capitalize First Letter of Each Word in a File. b.) Write a Python Program to Print the Contents of File in Reverse Order.	
13.	WAP to catch an exception and handle it using try and except code blocks.	
14.	Write a Python Program to Append, Delete and Display Elements of a List using Classes	
15.	Write a Python Program to Find the Area and Perimeter of the Circle using Class.	
16.	Create an interactive application using Python’s Tkinter library Fot graphics programming.	

Program 1

A. Write a Python Program to Calculate the Area of a Triangle.

Solution:

a=int(input("Base of Triangle="))

b=int(input("Height of Triangle="))

c=(a*b)/2

print("Area of the Triangle is",c)

Output:

```
In [5]: a=int(input("Base of Triangle="))
        b=int(input("Height of Triangle="))
        c=(a*b)/2
        print("Area of the Triangle is",c)

        Base of Triangle=4
        Height of Triangle=5
        Area of the Triangle is 10.0
```

B. Write a Python Program to Swap Two Variables.

Solution:

a=int(input("Value of A="))

b=int(input("Value of B="))

x=a

a=b

b=x

print("Value of A after swapping is",a)

print("Value of B after swapping is",b)

Output:

```
In [6]: a=int(input("Value of A="))
        b=int(input("Value of B="))
        x=a
        a=b
        b=x
        print("Value of A after swapping is",a)
        print("Value of B after swapping is",b)

        Value of A=5
        Value of B=7
        Value of A after swapping is 7
        Value of B after swapping is 5
```

C. Write a Python Program to Convert Celsius to Fahrenheit.

Solution:

a=int(input("Temperature in Celcius="))

b=(a*9/5)+32

print("Temperature in Fahrenheit =",b)

Output:

```
In [7]: a=int(input("Temperature in Celcius="))
        b=(a*9/5)+32
        print("Temperature in Fahrenheit =",b)

        Temperature in Celcius=65
        Temperature in Fahrenheit = 149.0
```

Program 2

A. Write a Python Program to Check if a Number is Odd or Even.

Solution:

```
a=int(input("Enter the Number:"))
b=a//2
if(b==1):
    print("The Number is Even.")
else:
    print("The Number is Odd.")
```

Output:

```
In [4]: a=int(input("Enter the Number:"))
        b=a//2
        if(b==1):
            print("The Number is Even.")
        else:
            print("The Number is Odd.")

        Enter the Number:5
        The Number is Odd.
```

B. Write a Python Program to Check if a Number is Positive, Negative or Zero.

Solution:

```
a=int(input("Enter the Number:"))
if(a<0):
```

```
    print("The Number is Negative.")
elif(a>0):
    print("The Number is Positive.")
else:
    print("Zero")
```

Output:

```
In [8]: a=int(input("Enter the Number:"))
        if(a<0):
            print("The Number is Negative.")
        elif(a>0):
            print("The Number is Positive.")
        else:
            print("Zero")

        Enter the Number:5
        The Number is Positive.
```

C. Write a Python Program to Check Armstrong Number.

Solution:

```
a = int(input("Enter a number: "))
sum = 0
temp = a
while temp > 0:
    digit = temp % 10
    sum += digit ** 3
    temp //= 10
if a == sum:
    print(a,"is an Armstrong number")
else:
    print(a,"is not an Armstrong number")
```

Output:

```
In [10]: a = int(input("Enter a number: "))
         sum = 0
         temp = a
         while temp > 0:
            digit = temp % 10
            sum += digit ** 3
            temp //= 10
         if a == sum:
            print(a,"is an Armstrong number")
         else:
            print(a,"is not an Armstrong number")
```

```
Enter a number: 153
153 is an Armstrong number
```

Program 3

A. Write a Python Program to check if a given number is Fibonacci number.

Solution:

```
x= int(input("Enter a Number: "))
a= 0
b=1
while b < x:
    a, b = b, a + b
if b == x:
    print(x, "is a Fibonacci number.")
else:
    print(x, "is not a Fibonacci number.")
```

Output:

```
In [12]: x= int(input("Enter a Number: "))
         a= 0
         b=1
         while b < x:
             a, b = b, a + b
         if b == x:
             print(x, "is a Fibonacci number.")
         else:
             print(x, "is not a Fibonacci number.")
```

```
Enter a Number: 5
5 is a Fibonacci number.
```

B. Write a Python Program to print cube sum of first n natural numbers.

Solution:

```
n=int(input("Enter value of n:"))
def cubesum(n):
  sum = 0
  for I in range(1, n+1):
    sum +=i*i*i
  return sum
print("The sum of cubes is:",cubesum(n))
```

Output:

```
In [13]: n=int(input("Enter value of n:"))
         def cubesum(n):
            sum = 0
            for i in range(1, n+1):
               sum +=i*i*i
            return sum
         print("The sum of cubes is:",cubesum(n))

         Enter value of n:5
         The sum of cubes is: 225
```

C. Write a Python Program to Write a Python program to print all odd numbers in a range.

Solution:

```
start = int(input("Enter the start of range:"))
end = int(input("Enter the end of range:"))

# iterating each number in list
for num in range(start, end + 1):

   # checking condition
   if num % 2 != 0:
       print(num)
S
```

Output:

```
In [14]: start = int(input("Enter the start of range:"))
         end = int(input("Enter the end of range:"))

         # iterating each number in list
         for num in range(start, end + 1):

             # checking condition
             if num % 2 != 0:
                 print(num)
```

```
Enter the start of range:4
Enter the end of range:8
5
7
```

Program 4

A. Write a Python Program to Print Pascal Triangle.

Solution:

```
from math import factorial
n=int(input("Enter value of n:"))
def print_pascal_triangle(n):
    for I in range(n):
        for j in range(n-i+1):
            print(end=" ")
        for j in range(i+1):
            print(factorial(i)//(factorial(j)*factorial(i-j)), end=" ")
        print()
print_pascal_triangle(n)
```

Output:

```
In [17]: from math import factorial
         n=int(input("Enter value of n:"))
         def print_pascal_triangle(n):
             for i in range(n):
                 for j in range(n-i+1):
                     print(end=" ")
                 for j in range(i+1):
                     print(factorial(i)//(factorial(j)*factorial(i-j)), end=" ")
                 print()
         print_pascal_triangle(n)
```

```
Enter value of n:4
     1
    1 1
   1 2 1
  1 3 3 1
```

B. WAP to Draw the following Pattern for n number:

Solution:

```
k=0
n=int(input("Enter value of n:"))
for I in range(n,0,-1):
   k=k+1
   for j in range(1,i+1):
      print(k,end=" ")
   print()
```

Output:

```
In [18]: k=0
         n=int(input("Enter value of n:"))
         for i in range(n,0,-1):
             k=k+1
             for j in range(1,i+1):
                 print(k,end=" ")
             print()

         Enter value of n:4
         1 1 1 1
         2 2 2
         3 3
         4
```

Program 5

Write a program with a function that accepts a string from keyboard and create a new string after converting character of each word capitalized.

Solution:

```
def capitalize_words(string):
    words = string.split()
    capitalized_words = [word.capitalize() for word in words]
    return " ".join(capitalized_words)

string = input("Enter a string: ")
capitalized_string = capitalize_words(string)
print(capitalized_string)
```

Output:

```
In [19]: def capitalize_words(string):
             words = string.split()
             capitalized_words = [word.capitalize() for word in words]
             return " ".join(capitalized_words)

         string = input("Enter a string: ")
         capitalized_string = capitalize_words(string)
         print(capitalized_string)

         Enter a string: stop and smell roses
         Stop And Smell Roses
```

Program 6

A. Write a program that accepts a list from user. The program should reverse the Content of list and display it. Do not use reverse () method.

Solution:

```
def reverse_list(lst):
    for I in range(len(lst)//2):
        lst[i], lst[-i-1] = lst[-i-1], lst[i]

lst = input("Enter a list: ").split()
lst = [int(i) for I in lst]
reverse_list(lst)
print(lst)
```

Output:

```
In [20]: def reverse_list(lst):
             for i in range(len(lst)//2):
                 lst[i], lst[-i-1] = lst[-i-1], lst[i]

         lst = input("Enter a list: ").split()
         lst = [int(i) for i in lst]
         reverse_list(lst)
         print(lst)

         Enter a list: 1 2 3 4 5
         [5, 4, 3, 2, 1]
```

B. Find and display the largest number of a list without using built-in function max (). Your program should ask the user to input values in list from keyboard.

Solution:

```
def find_largest_number(a):
    largest = a[0]
    for I in range(1, len(a)):
        if a[i] > largest:
            largest = a[i]
    return largest

a = input("Enter a list of numbers: ").split()
a = [int(i) for I in a]
largest_number = find_largest_number(a)
print(f"The largest number in the list is {largest_number}.")
```

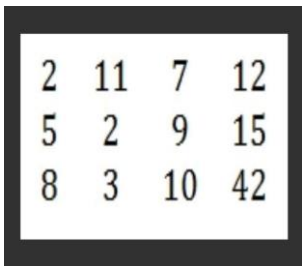
Output:

```
In [21]: def find_largest_number(a):
             largest = a[0]
             for i in range(1, len(a)):
                 if a[i] > largest:
                     largest = a[i]
             return largest

         a = input("Enter a list of numbers: ").split()
         a = [int(i) for i in a]
         largest_number = find_largest_number(a)
         print(f"The largest number in the list is {largest_number}.")

         Enter a list of numbers: 1 2 3 4 5
         The largest number in the list is 5.
```

```
Enter a list of numbers: 1 2 3 4 5
The largest number in the list is 5.
```

Program 7

Find the sum of each row of matrix of size m x n. For example, for the following matrix output will be like this:



Sum of row 1 = 32

Sum of row 2 = 31

Sum of row 3 = 63

Solution:

```
matrix = [
    [10, 5, 8, 9],
    [4, 12, 6, 9],
    [3, 7, 5, 48]
]
m = len(matrix)
n = len(matrix[0])
row_sums = []
for i in range(m):
    row_sum = sum(matrix[i])
    row_sums.append(row_sum)
    print(f"Sum of row {i+1} = {row_sum}")
```

Output:

```
In [1]: matrix = [
            [10, 5, 8, 9],
            [4, 12, 6, 9],
            [3, 7, 5, 48]
        ]
        m = len(matrix)
        n = len(matrix[0])
        row_sums = []
        for i in range(m):
            row_sum = sum(matrix[i])
            row_sums.append(row_sum)
            print(f"Sum of row {i+1} = {row_sum}")

        Sum of row 1 = 32
        Sum of row 2 = 31
        Sum of row 3 = 63
```

Sum of row 1 = 32
Sum of row 2 = 31
Sum of row 3 = 63

Program 8

a) Write a program that reads a string from keyboard and display:
*** The number of uppercase letters in the string.**
*** The number of lowercase letters in the string.**
*** The number of digits in the string.**
*** The number of whitespace characters in the string.**

Solution:

```
n=input("enter a string:")
u=0
l=0
d=0
w=0
for i in n:
    if i.isupper():
        u=u+1
    elif i.islower():
        l=l+1
    elif i.isdigit():
        d=d+1
    elif (i==" "):
        w=w+1
print("number of uppercase {} and number of lowercase{} and number of digit{} and
number of whitespace{}".format(u,l,d,w))
```

```
In [4]: n=input("enter a string:")
        u=0
        l=0
        d=0
        w=0
        for i in n:
            if i.isupper():
                u=u+1
            elif i.islower():
                l=l+1
            elif i.isdigit():
                d=d+1
            elif (i==" "):
                w=w+1
        print("number of uppercase {} and number of lowercase{} and number of digit{} and number of whitespace{}".format(u,l,d,w))

        enter a string:Welcome To Chitkara University
        number of uppercase 4 and number of lowercase23 and number of digit0 and number of whitespace3
```

b) Python Program to Find Common Characters in Two Strings.

Solution:

```
s1=input("enter string 1:")
s2=input("enter string 2:")
comm=set(s1)&set(s2)
print(comm)
```

Output:

```
In [5]: s1=input("enter string 1:")
        s2=input("enter string 2:")
        comm=set(s1)&set(s2)
        print(comm)

        enter string 1:Rivanshi
        enter string 2:Ritika
        {'i', 'R', 'a'}
```

c) Python Program to Count the Number of Vowels in a String.

Solution:

```
text=input("text:")
count=0
for character in text:
    if (character in "AaEeIiOoUu"):
        count=count+1
print("count:",count)
```

Output:

```
In [6]: text=input("text:")
        count=0
        for character in text:
            if (character in "AaEeIiOoUu"):
                count=count+1
        print("count:",count)

        text:I Love Pyhton
        count: 4
```

Program 9

a) **Write a Python program to check if a specified element presents in a tuple of tuples.**
Original list: (('Red' ,'White' , 'Blue'),('Green', 'Pink' , 'Purple'), ('Orange', 'Yellow', 'Lime'))
Check if White present in said tuple of tuples!
True
Check if Olive present in said tuple of tuples!
False

Solution:

```
tupple= ('Red' , 'Blue','Green', 'Pink' , 'Purple', 'Orange', 'Yellow', 'Lime')
if 'white' in tupple:
   print("yes")
else:
   print("no")
```

```
In [8]: tupple= ('Red' , 'Blue','Green', 'Pink' , 'Purple', 'Orange', 'Yellow', 'Lime')
        if 'white' in tupple:
            print("yes")
        else:
            print("no")

        no
```

b) Write a Python program to remove an empty tuple(s) from a list of tuples.
Sample data: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
Expected output: [('',), ('a', 'b'), ('a', 'b', 'c'), 'd']

Solution:

```
L= [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
L= [t for t in L if t]
print(L)
```

```
In [9]: L= [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
        L= [t for t in L if t]
        print(L)

        [('',), ('a', 'b'), ('a', 'b', 'c'), 'd']
```

Program 10

Write a Program in Python to Find the Differences Between Two Lists Using Sets.

Solution:

```
L1=[1,3,5,7,9]
L2=[1,2,4,6,7,8]
diff1=list(set(L1)-set(L2))
diff2=list(set(L2)-set(L1))
total=diff1+diff2
print(total)
```

```
In [10]: L1=[1,3,5,7,9]
         L2=[1,2,4,6,7,8]
         diff1=list(set(L1)-set(L2))
         diff2=list(set(L2)-set(L1))
         total=diff1+diff2
         print(total)

         [9, 3, 5, 8, 2, 4, 6]
```

Program 11

a) Write a Python program Remove duplicate values across Dictionary Values.

Input : test_dict = {'Manjeet': [1], 'Akash': [1, 8, 9]}
Output : {'Manjeet': [], 'Akash': [8, 9]}
Input : test_dict = {'Manjeet': [1, 1, 1], 'Akash': [1, 1, 1]}
Output : {'Manjeet': [], 'Akash': []}

Solution:

```
def remove_duplicate_values(dictionary):
    value_count = {}
    for values in dictionary.values():
        for value in values:
            if value in value_count:
                value_count[value] += 1
            else:
                value_count[value] = 1
    for key, values in dictionary.items():
        unique_values = [value for value in values if value_count[value] == 1]
        dictionary[key] = unique_values
    return dictionary
# Test cases
test_dict1 = {'Manjeet': [1], 'Akash': [1, 8, 9]}
result1 = remove_duplicate_values(test_dict1)
print(result1)
test_dict2 = {'Manjeet': [1, 1, 1], 'Akash': [1, 1, 1]}
result2 = remove_duplicate_values(test_dict2)
print(result2)
```

```
In [13]: def remove_duplicate_values(dictionary):
             value_count = {}
             for values in dictionary.values():
                 for value in values:
                     if value in value_count:
                         value_count[value] += 1
                     else:
                         value_count[value] = 1

             for key, values in dictionary.items():
                 unique_values = [value for value in values if value_count[value] == 1]
                 dictionary[key] = unique_values
             return dictionary
         # Test cases
         test_dict1 = {'Manjeet': [1], 'Akash': [1, 8, 9]}
         result1 = remove_duplicate_values(test_dict1)
         print(result1)

         test_dict2 = {'Manjeet': [1, 1, 1], 'Akash': [1, 1, 1]}
         result2 = remove_duplicate_values(test_dict2)
         print(result2)
```

```
{'Manjeet': [], 'Akash': [8, 9]}
{'Manjeet': [], 'Akash': []}
```

b) Write a Python program to Count the frequencies in a list using dictionary in Python.
Input : [1, 1, 1, 5, 5, 3, 1, 3, 3, 1,4, 4, 4, 2, 2, 2, 2]
Output :

1 : 5
2 : 4
3 : 3
4 : 3
5 : 2

Solution:

```
def count_frequencies(input_list):
    frequency_dict = {}

    for element in input_list:
        if element in frequency_dict:
            frequency_dict[element] += 1
        else:
            frequency_dict[element] = 1
    return frequency_dict
input_list=[1,1,1,5,5,3,1,3,3,1,4,4,4,2,2,2,2]
result=count_frequencies(input_list)
print(result)
```

```
In [2]: def count_frequencies(input_list):
            frequency_dict = {}

            for element in input_list:
                if element in frequency_dict:
                    frequency_dict[element] += 1
                else:
                    frequency_dict[element] = 1
            return frequency_dict
        input_list=[1,1,1,5,5,3,1,3,3,1,4,4,4,2,2,2,2]
        result=count_frequencies(input_list)
        print(result)

        {1: 5, 5: 2, 3: 3, 4: 3, 2: 4}
```

Program 12

a) Write a Python Program to Capitalize First Letter of Each Word in a File.

Solution:

```
f=open('file.txt','r')
output=""
for line in f:
   output+=line.title()
   print(output)
f=open("output.txt","w")
f.write(output)
```

```
Hi I am Rivanshi
I am here to solve the python problems
```

b) Write a Python Program to Print the Contents of File in Reverse Order.

Solution:

```
file=input("enter location ,name ,extension of file:")
for line in reversed(list(open(file))):
      print(line.rstrip())
```

```
Bye
Hello
```

Problem 13

Write a program to catch an exception and handle it using try and except code block.

Solution:

```
try:
   num1=int(input("Enter a number:"))
   num2=int(input("Enter another number:"))
   result=num1/num2
   print("Result:",result)
except ZeroDivisionError:
   print("Error:Division by zero not allowed")
except ValueError:
   print("Error:Enter valid numbers")
except Exception as e:
   print("An unexpected error",str(e))
```

```
In [3]: try:
            num1=int(input("Enter a number:"))
            num2=int(input("Enter another number:"))
            result=num1/num2
            print("Result:",result)
        except ZeroDivisionError:
            print("Error:Division by zero not allowed")
        except ValueError:
            print("Error:Enter valid numbers")
        except Exception as e:
            print("An unexpected error",str(e))

Enter a number:67
Enter another number:0
Error:Division by zero not allowed
```

Program 14

Write a Python Program to Append, Delete and Display Elements of a List using Classes.

Solution:

```
class MyList:
    def __init__(self):
        self.n = []
    def add(self, a):
        return self.n.append(a)
    def remove(self, b):
        self.n.remove(b)
    def display(self):
        return (self.n)
obj = MyList()
choice = 1
while choice != 0:
    print("0. Exit")
    print("1. Add")
    print("2. Delete")
    print("3. Display")
    choice = int(input("Enter choice: "))
    if choice == 1:
        n = int(input("Enter number to append: "))
        obj.add(n)
        print("List: ", obj.display())
    elif choice == 2:
        n = int(input("Enter number to remove: "))
        obj.remove(n)
        print("List: ", obj.display())
    elif choice == 3:
        print("List: ", obj.display())
    elif choice == 0:
        print("Exiting!")
    else:
        print("Invalid choice!!")
```

In [6]:
```
class MyList:
    def __init__(self):
        self.n = []
    def add(self, a):
        return self.n.append(a)
    def remove(self, b):
        self.n.remove(b)
    def display(self):
        return (self.n)
obj = MyList()
choice = 1
while choice != 0:
    print("0. Exit")
    print("1. Add")
    print("2. Delete")
    print("3. Display")
    choice = int(input("Enter choice: "))
    if choice == 1:
        n = int(input("Enter number to append: "))
        obj.add(n)
        print("List: ", obj.display())
    elif choice == 2:
        n = int(input("Enter number to remove: "))
        obj.remove(n)
        print("List: ", obj.display())
    elif choice == 3:
        print("List: ", obj.display())
    elif choice == 0:
        print("Exiting!")
    else:
        print("Invalid choice!!")
```

```
0. Exit
1. Add
2. Delete
3. Display
Enter choice: 3
List:  []
0. Exit
1. Add
2. Delete
3. Display
Enter choice: 32
Invalid choice!!
0. Exit
1. Add
2. Delete
3. Display
Enter choice: 0
Exiting!
```

Program 15

Write a Python Program to Find the Area and Perimeter of the Circle using Class

Solution:

```
import math
class circle_compute():
   def __init__(self,my_radius):
      self.radius=my_radius
   def area_calculate(self):
      return math.pi*(self.radius**2)
   def perimeter_calculate(self):
      return 2*math.pi*self.radius
my_result = int(input("Enter the radius of circle..."))
my_instance = circle_compute(my_result)
print("The radius entered is :")
print(my_result)
print("The computed area of circle is ")
print(round(my_instance.area_calculate(),2))
print("The computed perimeter of circle is :")
print(round(my_instance.perimeter_calculate(),2))
```

```
In [8]: import math
        class circle_compute():
            def __init__(self,my_radius):
                self.radius=my_radius
            def area_calculate(self):
                return math.pi*(self.radius**2)
            def perimeter_calculate(self):
                return 2*math.pi*self.radius
        my_result = int(input("Enter the radius of circle..."))
        my_instance = circle_compute(my_result)
        print("The radius entered is :")
        print(my_result)
        print("The computed area of circle is ")
        print(round(my_instance.area_calculate(),2))
        print("The computed perimeter of circle is :")
        print(round(my_instance.perimeter_calculate(),2))
```

```
Enter the radius of circle...4
The radius entered is :
4
The computed area of circle is
50.27
The computed perimeter of circle is :
25.13
```

Program 16

Create an interactive application using Python's Tkinter library Fot graphics programming

Solution :

```
import tkinter as tk
def change_color():
   new_color = "#" + hex(randint(0, 255))[2:] * 3
   root.configure(bg=new_color)
root = tk.Tk()
root.title("Color Changer")
button = tk.Button(root, text="Change Color", command=change_color)
button.pack()
# Start the Tkinter event loop
root.mainloop()
```

```python
import tkinter as tk

def change_color():
    # Randomly choose a new color
    new_color = "#" + hex(randint(0, 255))[2:] * 3
    root.configure(bg=new_color)

# Create the main window
root = tk.Tk()
root.title("Color Changer")

# Create a button
button = tk.Button(root, text="Change Color", command=change_color)
button.pack()

# Start the Tkinter event loop
root.mainloop()
```