DEPARTMENT OF COMPUTER APPLICATION

TKM COLLEGE OF ENGINEERING

KOLLAM – 691005

20MCA131 - PROGRAMMING LAB

PRACTICAL RECORD BOOK

First Semester MCA

2020-2021

Submitted by:

NAME : ANANYA B

ROLL NO : MCA113

DEPARTMENT OF COMPUTER APPLICATION

TKM COLLEGE OF ENGINEERING

KOLLAM – 691005

Certificate

This is a bonafide record of the work done by ANANYA B in the First Semester in Programming Lab Course(20MCA131) towards the partial fulfillment of the degree of Master of Computer Applications during the academic year 2020-2021.

Staff Member in-charge

Examiner

………………………..

………………………..

INDEX

COURSE OUTCOME 1 - Understands basics of Python Programming language including input/output functions, operators, basic and collection data types

COURSE OUTCOME 2 - Implement decision making, looping constructs and functions

COURSE OUTCOME 3 - Design modules and packages - built in and user defined packages

COURSE OUTCOME 4 - Implement object-oriented programming and exception handling.

COURSE OUTCOME 5 - Create files and form regular expressions for effective search operations on strings and files.

COURSE OUTCOME 1

PROGRAM NO: 1

AIM: Display future leap years from current year to a final year entered by user.

ALGORITHM:

Step1: Take current year and final year as inputs

Step2: if current year < final year

Step3: Check for leap year condition

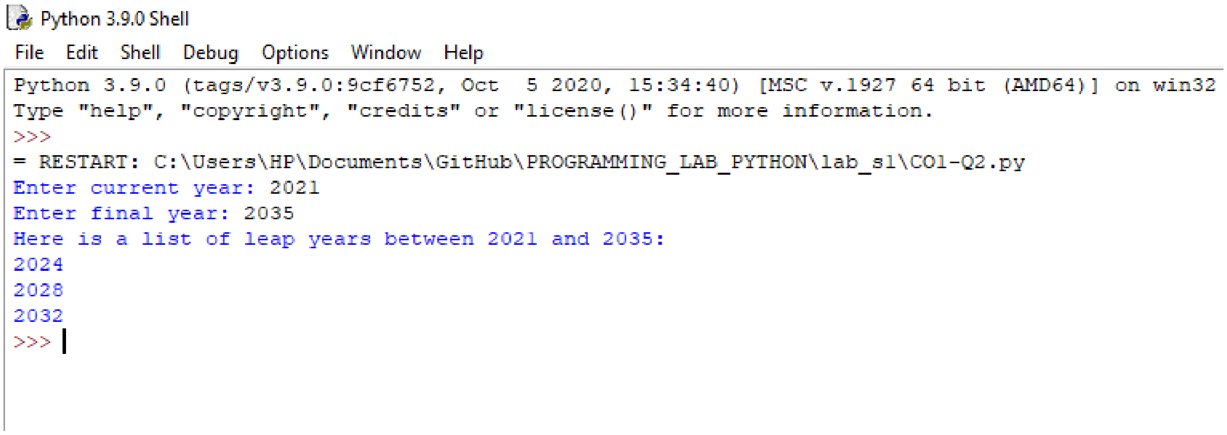
Step4: Then print list of leap years between current and final year

PROGRAM:

```
current = int(input("Enter current year: "))
final = int(input("Enter final year: "))

if current < final:
print ("Here is a list of leap years between " + str(current) + " and " + str(final) + ":")

while current < final:
if current % 4 == 0:
print(current)
if current % 100 == 0 and current % 400 == 0:
print(current)
current += 1
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT:



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q2.py
Enter current year: 2021
Enter final year: 2035
Here is a list of leap years between 2021 and 2035:
2024
2028
2032
>>> 
```

PROGRAM NO: 2

AIM: 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 word

(d) List ordinal value of each element of a word (Hint: use ord() to get ordinal values)

ALGORITHM:

a) Step1: Take in the number of elements to be in the list from the user.
Step2: Using a for in list comprehensions, get the elements one by one from the list and check if it is positive
Step3: If it is positive, print the numbers as a list

b) Step1: Take input N
Step2: Compute square of numbers upto range N using list comprehension
Step3: Take result as list and display

c)Step1: Give list of vowels as V

Step2: Take a word as input

Step3: Check the letters in word and compare with list of vowels in V

Step4: If found ,then take that letters as resultant list and display

d)Step1: Take a word as input

Step2: And make that word as list

Step3: Using list comprehension and ord() function find ordinal value of each letter

Step4: Place that result as list and display it

PROGRAM:

A)

```
 list1 = [1,-1, -21, 0, 45, 66,2,-3,4,-6, -93]
print(list1)
a=[num for num in list1 if num>=0]
print("Positive integers in the list are:",a)
```

B)

```
N=int(input("Enter limit N:"))
x = (x**2 for x in range(N))
x = list(x)
print(x)
```

C)

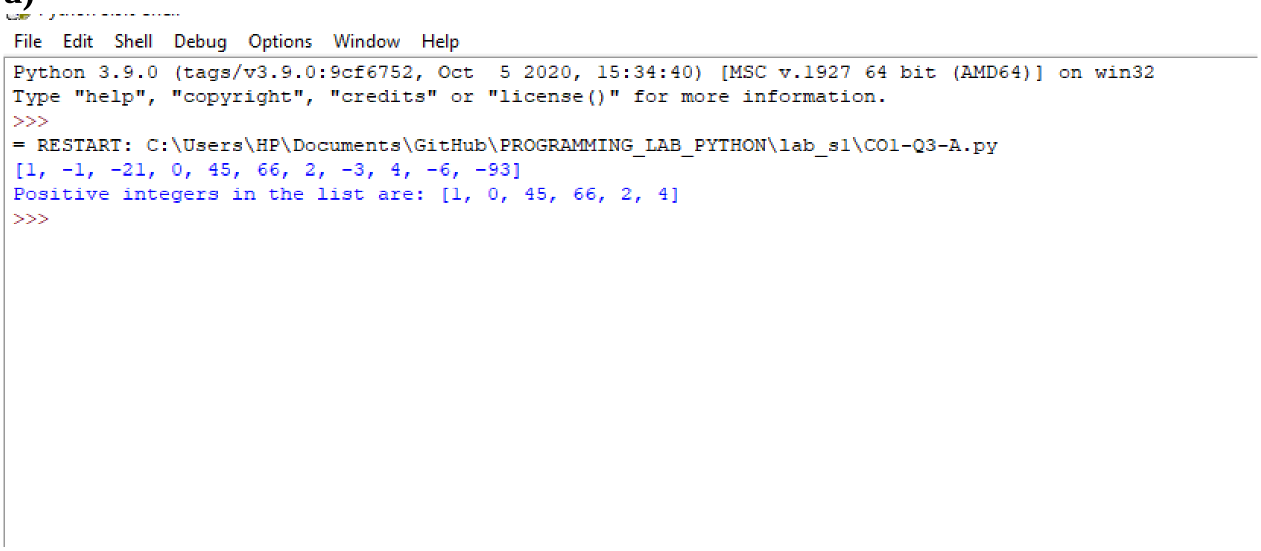
```
V =[' a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U' ]
print("V=['a', 'e', 'i', 'o', 'u", ‘A', 'E', 'I', 'O', 'U' ]")

w=str(input("Enter the word: "))

x = [x for x in w if any([v in x for v in V])]
x = list(x)
print ("Vowels in given word:",x)
```

```
    D)a=str(input("Enter
    word:")) a=list(a)
x=[ord(x) for x in a ]
x=list(x)
print(x)
```

RESULT: The above program is successfully executed and obtained the outpu

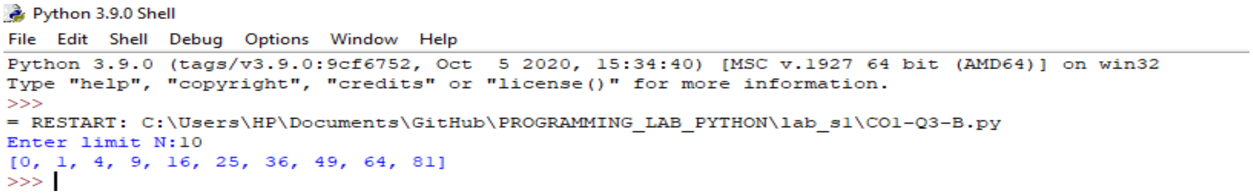
OUTPUT:

a)



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q3-A.py
[1, -1, -21, 0, 45, 66, 2, -3, 4, -6, -93]
Positive integers in the list are: [1, 0, 45, 66, 2, 4]
>>> 
```

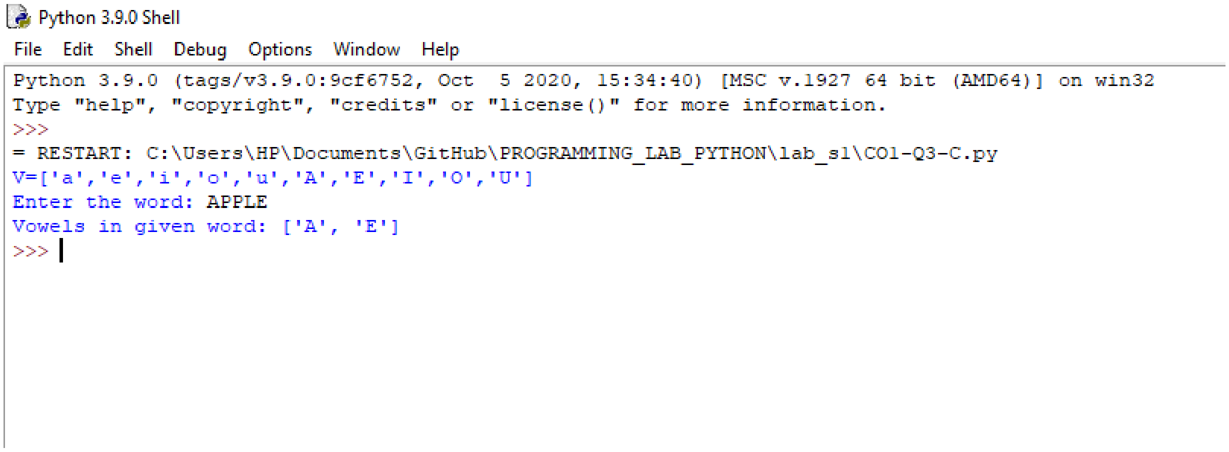
b)



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q3-B.py
Enter limit N:10
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
>>> 
```

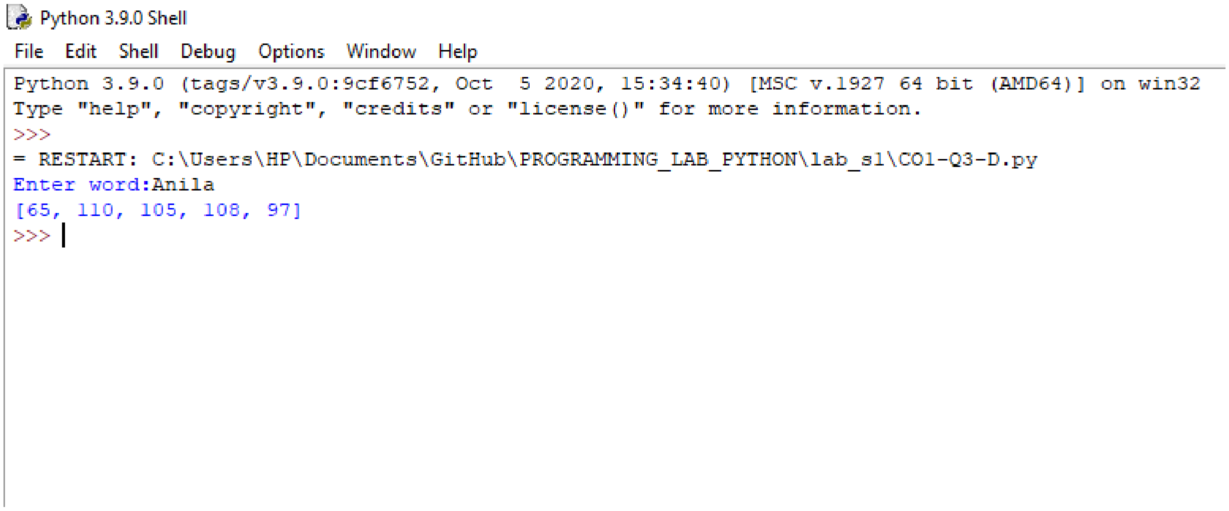
s

c)



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q3-C.py
V=['a','e','i','o','u','A','E','I','O','U']
Enter the word: APPLE
Vowels in given word: ['A', 'E']
>>> 
```

d)



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q3-D.py
Enter word:Anila
[65, 110, 105, 108, 97]
>>> 
```

PROGRAM NO: 3

AIM: Count the occurrences of each word in a line of text.

ALGORITHM:

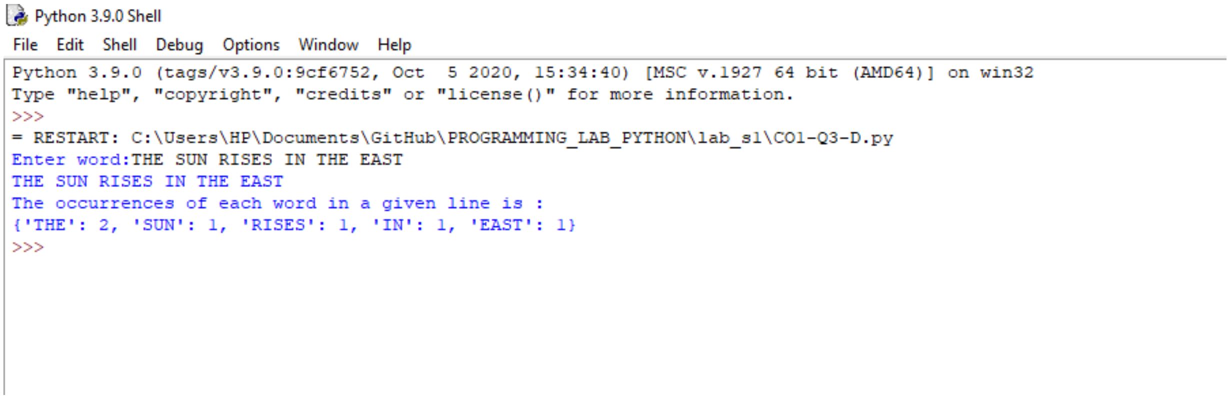
```
Step1: Take a string as input a
Step2: Then splitted that string using split() function
Step3: For each word in string count gets incrementes
Step4: Display the count value as occurences of each word
```

PROGRAM:

```
a=str(input("Enter word:"))
print(a)
s=a.split(' ')
count = {}
for n in s:
   count[n]=count.get(n,0)+1
print("The occurrences of each word in a given line is :")
print(count)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q3-D.py
Enter word:THE SUN RISES IN THE EAST
THE SUN RISES IN THE EAST
The occurrences of each word in a given line is :
{'THE': 2, 'SUN': 1, 'RISES': 1, 'IN': 1, 'EAST': 1}
>>> 
```

PROGRAM NO: 4

AIM: Prompt the user for a list of integers. For all values greater than 100, store ‘over’ instead.

ALGORITHM:

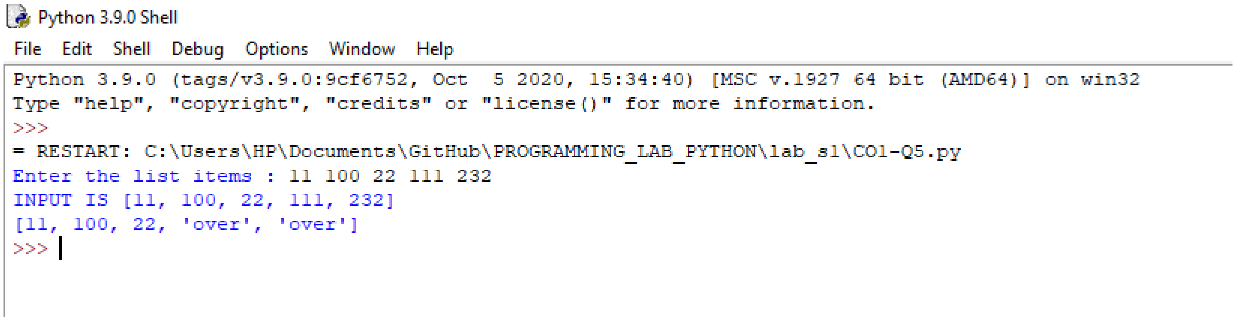
```
Step1: Start
Step2: Initialize an empty list
Step3: Then give list items as input by user
Step4: Using list comprehension check for each value is greater than 100
Step5: If greater then 100 ,the change that value with word ‘over’
Step6: Displayed the results as a list elements
Step7:Stop
```

PROGRAM:

```
lst = [ ]
lst = [int(item) for item in input("Enter the list items : ").split()]
print("INPUT IS",lst)
x= ["over" if x>100 else x for x in lst]
lst=list(x)
print(lst)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q5.py
Enter the list items : 11 100 22 111 232
INPUT IS [11, 100, 22, 111, 232]
[11, 100, 22, 'over', 'over']
>>> 
```

PROGRAM NO: 5

AIM: Store a list of first names. Count the occurrences of ‘a’ within the list

ALGORITHM:

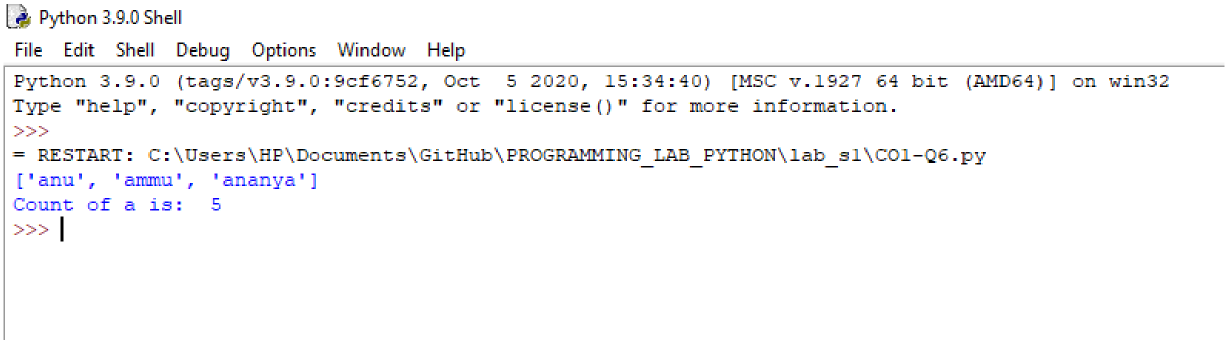
```
Step1: Take inputs as list of names having letter ‘a’
Step2: initialize i and count value as 0
Step3: check for the occurrence of letter a in those names
Step4: if found the count gets incremented
Step5: And displayed the final count value as result of occurrence
```

PROGRAM:

```
lst=['anu','ammu','ananya']
print(lst)
i=0
count=0
while i<len(lst):
        count=count+lst[i].count('a')
        i=i+1

print("Count of a is: " ,count)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q6.py
['anu', 'ammu', 'ananya']
Count of a is:  5
>>> 
```

PROGRAM NO: 6

AIM: Enter 2 lists of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both

ALGORITHM:

```
Step1: Start
Step2: Take two lists of numbers lst1 and lst2 as inputs and displayed it
Step3: Then find the length of each list using len()
Step4: And check whether they are of same length or not
Step5: If both lists are of same length then print “SAME LENGTH”
        Else print “NOT SAME LENGTH”
Step6: Find the sum of each list and also check both sum are equal
Step7: If same print “SUM IS SAME”
        Else print “SUM IS NOT SAME”
Step7: Then find the common elements in the list using intersection () in sets by converting both lists into sets
Step9: Then display that common elements as result as a list
Step10: Stop
```

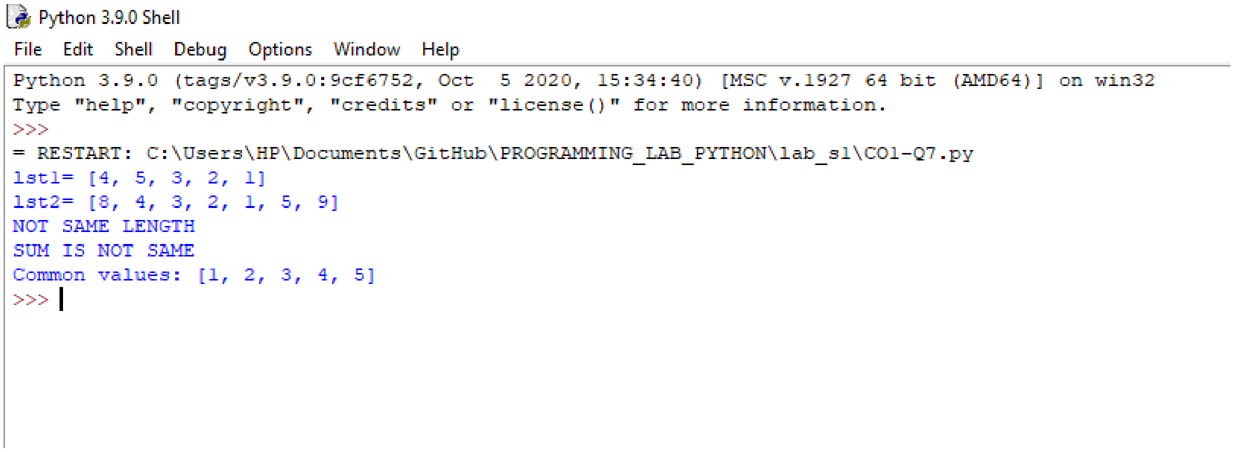
PROGRAM:

```
lst1=[4,5,3,2,1]

lst2=[8,4,3,2,1,5,9]

print("lst1=",lst1)

print("lst2=",lst2)

a=len(lst1)

b=len(lst2)

if a==b:

    print("SAME LENGTH")
```

```
else:
    print("NOT SAME LENGTH")
s1=sum(lst1)
s2=sum(lst2)
if s1==s2:
    print("SUM IS SAME")
else:
        print("SUM IS NOT SAME")
lst1=set(lst1)
lst2=set(lst2)
i = lst1.intersection(lst2)
i=list(i)
print("Common values:",i)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q7.py
lst1= [4, 5, 3, 2, 1]
lst2= [8, 4, 3, 2, 1, 5, 9]
NOT SAME LENGTH
SUM IS NOT SAME
Common values: [1, 2, 3, 4, 5]
>>> 
```

PROGRAM NO: 7

AIM: Get a string from an input string where all occurrences of first character replaced with ‘$’, except first character.

ALGORITHM:

Step1: Start

Step2: Take string as input str1

Step3: Place the first character of str1 into char

Step4: Then replace occurrence the first character in the string using replace method

Step5: Then append the value at char and the string from position 1

Step6: Displays the appended string as new string

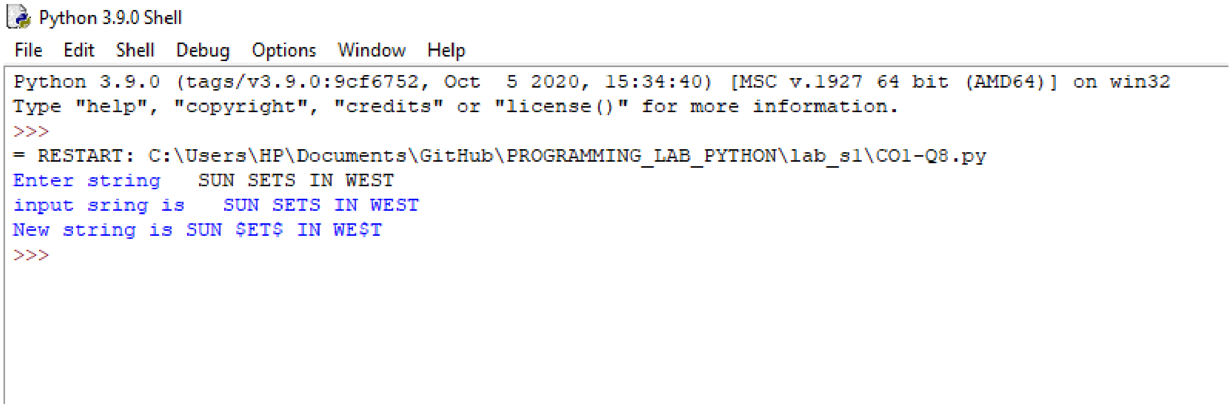
Step7: Stop

PROGRAM:

```
str1=input('Enter string ')
print('input sring is ',str1)
char = str1[0]
str1 = str1.replace(char, '$')
str1 = char + str1[1:]
print('New string is',str1)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q8.py
Enter string   SUN SETS IN WEST
input sring is   SUN SETS IN WEST
New string is SUN $ET$ IN WE$T
>>> 
```

PROGRAM NO: 8

AIM: Create a string from given string where first and last characters exchanged.

ALGORITHM:

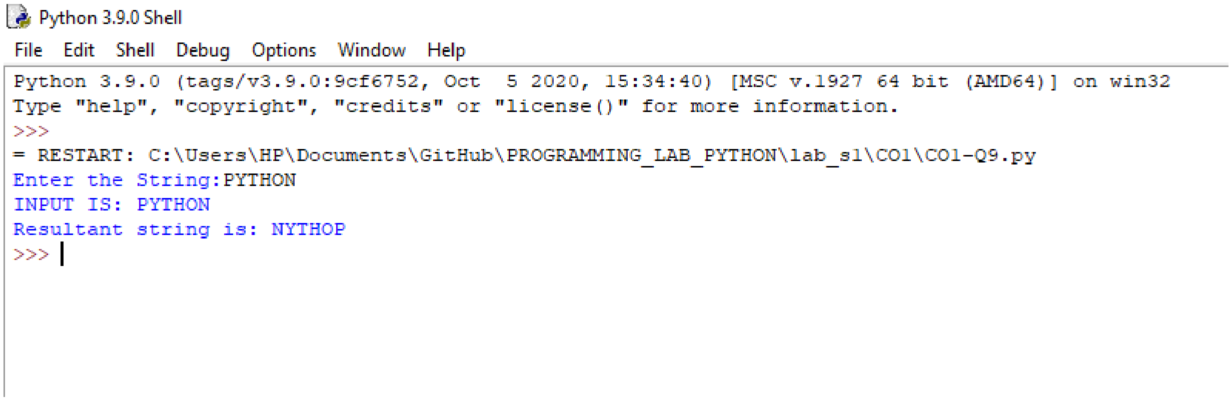
```
Step1: Start
Step2: Take input as a string ‘s’
Step3: Slice the string into 3 parts
Step4: One from first position to second last as slice_mid
Step5: And slice_beg at position 0 and at position last as slice_end
Step6: Then concatenate the slice_end, slice_mid, slice_beg as output
Step7: Stop
```

PROGRAM:

```
s = str(input("Enter the String:"))
print("INPUT IS:",s)
slice_mid=s[1:-1]
slice_beg=s[0]
slice_end=s[-1:]
print("Resultant string is:",slice_end+slice_mid+slice_beg)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1\CO1-Q9.py
Enter the String:PYTHON
INPUT IS: PYTHON
Resultant string is: NYTHOP
>>> 
```

PROGRAM NO: 9

AIM: Accept the radius from user and find area of circle.

ALGORITHM:

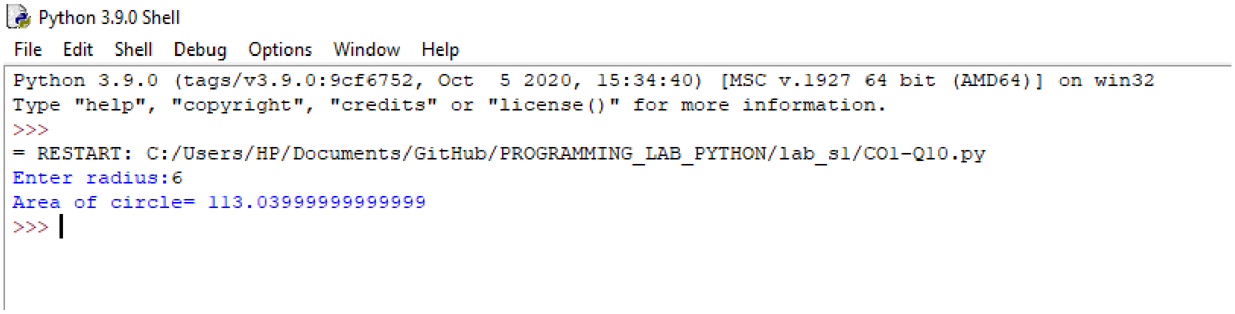
```
Step1: Start
Step2: Take R as radius of circle as input and initialize pi =3.14
Step3: Then print Area of circle as pi*R*R
Step4:Stop
```

PROGRAM:

```
pi=3.14
R=int(input("Enter radius:"))
print("Area of circle=",pi*R*R)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:/Users/HP/Documents/GitHub/PROGRAMMING_LAB_PYTHON/lab_s1/CO1-Q10.py
Enter radius:6
Area of circle= 113.03999999999999
>>> 
```

PROGRAM NO: 10

AIM: Find biggest of 3 numbers entered.

ALGORITHM:

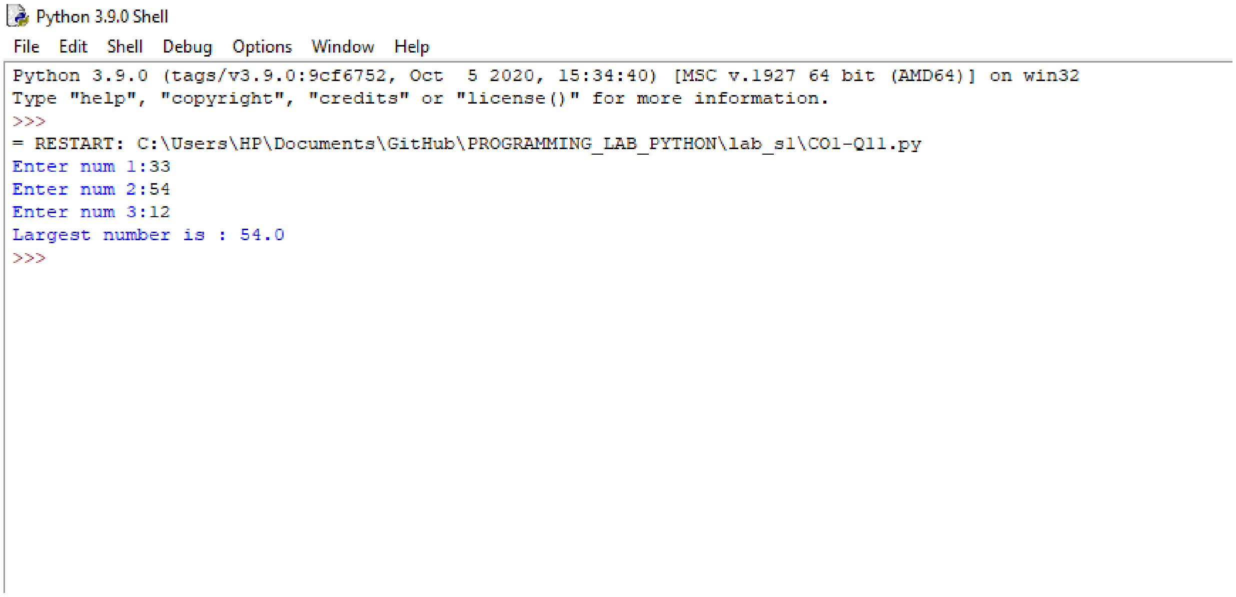
```
Step1: Start
Step2: Take n1,n2,n3 as three input numbers
Step3: Check which is large using max() function
Step4: And display the result as output
Step5 : Stop
```

PROGRAM:

```
n1 = float(input("Enter num 1:"))
n2 = float(input("Enter num 2:"))
n3 = float(input("Enter num 3:"))
large=max(n1 , n2 , n3)
print("Largest number is :",large)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q11.py
Enter num 1:33
Enter num 2:54
Enter num 3:12
Largest number is : 54.0
>>> 
```

PROGRAM NO: 11

AIM: Accept a file name from user and print extension of that

ALGORITHM:

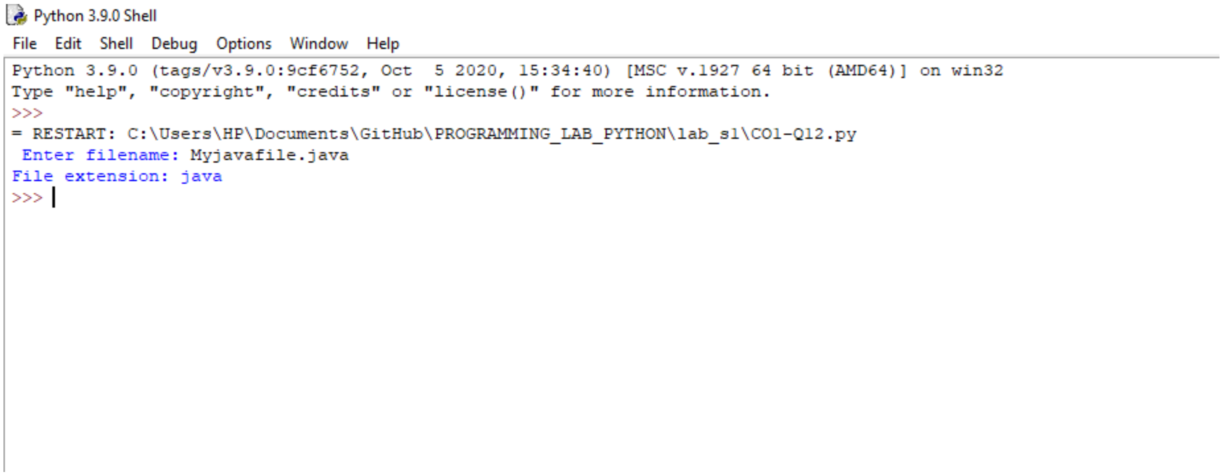
```
Step1: Start
Step2: Take input as a filename
Step3: Then split the filename on the occurrances of ‘.’
Step4: Then save the file extension in ‘t’
Step5: Display that extension as output
Step6: Stop
```

PROGRAM:

```
str=input(" Enter filename: ")
t=str.split('.')
print("File extension: " +t[-1])
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q12.py
 Enter filename: Myjavafile.java
File extension: java
>>> 
```

PROGRAM NO: 12

AIM: Create a list of colors from comma-separated color names entered by user. Display first and last colors.

ALGORITHM:

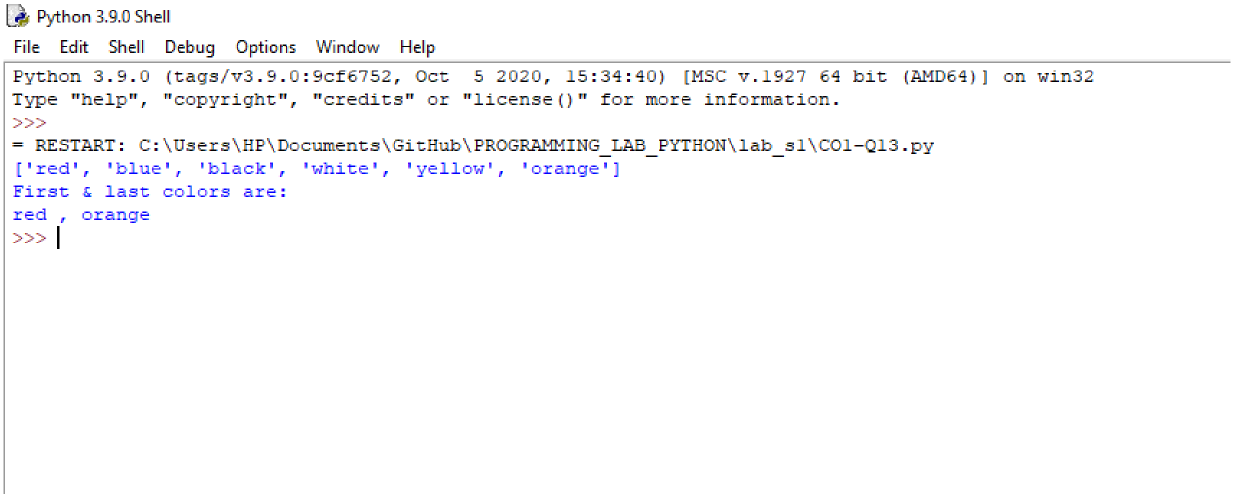
```
Step1: Start
Step2: Take a list of colors as input
Step3: Take the colors at position first and last and place it in a and b
Step4: Then display those colors as output
Step5: Stop
```

PROGRAM:

```
color_lst=["red","blue","black","white","yellow","orange"]
print(color_lst)
a = color_lst[0]
b= color_lst[-1]
print("First & last colors are:")
print(a,b, sep = " , ")
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q13.py
['red', 'blue', 'black', 'white', 'yellow', 'orange']
First & last colors are:
red , orange
>>> 
```

PROGRAM NO: 13

AIM: Accept an integer n and compute n+nn+nnn.

ALGORITHM:

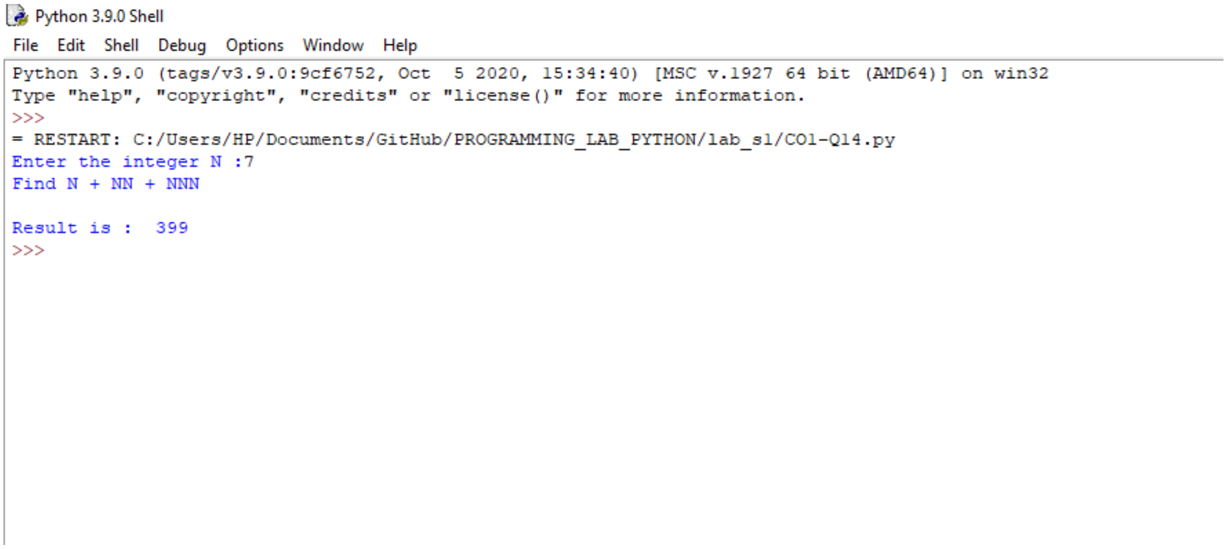
```
Step1: Start
Step2: Take an integer N as input
Step3: Initialize tmp=N
Step4: Then created tmp1 and tmp2 to compute n*n and n*n*n terms
Step5: And take the output to variable cmp and displayed it as result
Step6: Stop
```

PROGRAM:

```
N = int(input("Enter the integer N :"))
tmp = N
tmp1 = tmp*tmp
tmp2 = tmp*tmp*tmp
print("Find N + NN + NNN\n")
comp = tmp + tmp1 + tmp2
print("Result is : ",comp)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:/Users/HP/Documents/GitHub/PROGRAMMING_LAB_PYTHON/lab_s1/CO1-Q14.py
Enter the integer N :7
Find N + NN + NNN

Result is :  399
>>> 
```

PROGRAM NO: 14

AIM: Print out all colors from color-list1 not contained in color-list2

ALGORITHM:

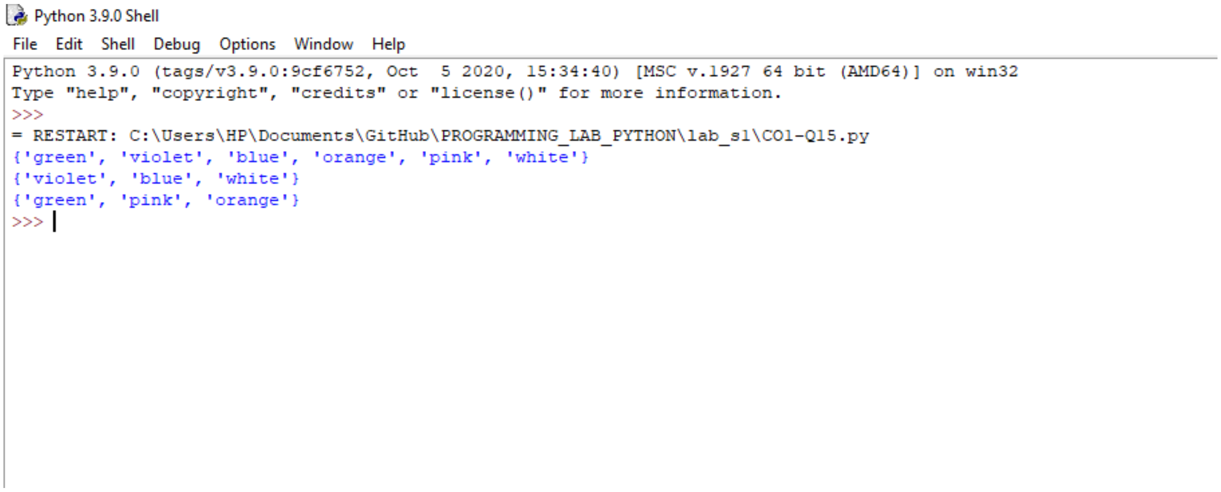
```
Step1: Start
Step2: Take 2 set of colors as input
Step3: Using difference () of sets to find the colors not in color-list 2
Step4: And place the result in variable ‘a’
Step5: Then print ‘a’
Step6: Stop
```

PROGRAM:

```
colorlist1=set(['orange','green','blue','violet','pink','white'])
print(colorlist1)
colorlist2=set(['white','blue','violet'])
print(colorlist2)
a=(colorlist1.difference(colorlist2))
print(a)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q15.py
{'green', 'violet', 'blue', 'orange', 'pink', 'white'}
{'violet', 'blue', 'white'}
{'green', 'pink', 'orange'}
>>> 
```

PROGRAM NO: 15

AIM: Create a single string separated with space from two strings by swapping the character at position 1.

ALGORITHM:

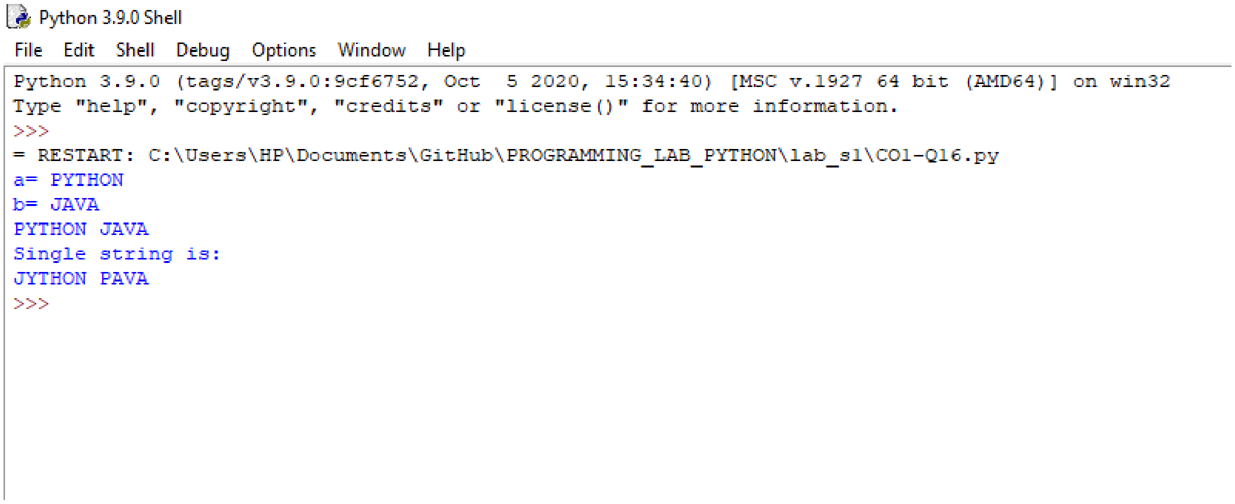
```
Step1: Start
Step2: Take 2 strings as inputs
Step3: Swap the first characters of those strings
Step4: Then concatenate those strings with space after swap
Step5: Display that single string as output
Step6: Stop
```

PROGRAM:

```
a= "PYTHON"
b="JAVA"
print ("a=",a)
print("b=",b)
print(a +" " + b )
a1 = b[:1] + a[1:]
b2= a[:1] + b[1:]
print("Single string is:")
print(a1 + " "+ b2)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q16.py
a= PYTHON
b= JAVA
PYTHON JAVA
Single string is:
JYTHON PAVA
>>> 
```

PROGRAM NO: 16

AIM: Sort dictionary in ascending and descending order.

ALGORITHM:

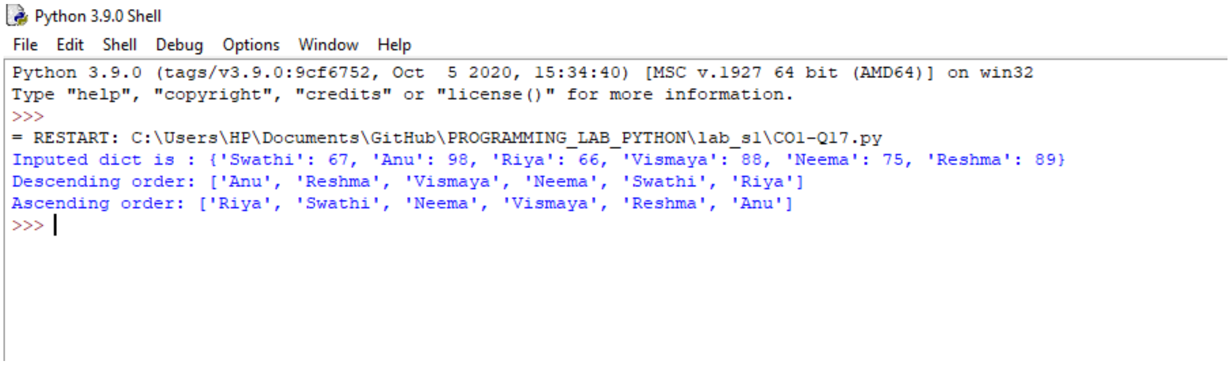
```
Step1: Start
Step2: Inputed a dictionary a1
Step3: a1 is sorted ascendingly using sorted() function
Step4: a1 is descendingly sorted using sorted() function with reverse=True
Step5: Displayed that two outputs
Step6:Stop
```

PROGRAM:

```
a1 ={'Swathi':67,'Anu':98,'Riya':66,'Vismaya':88,'Neema':75,'Reshma':89}
print("Inputed dict is :", a1)
a1_sorted_keys = sorted(a1, key=a1.get, reverse=True)
a1_sorted_keys_2 = sorted(a1, key=a1.get)

print("Descending order:",a1_sorted_keys)
print("Ascending order:",a1_sorted_keys_2)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q17.py
Inputed dict is : {'Swathi': 67, 'Anu': 98, 'Riya': 66, 'Vismaya': 88, 'Neema': 75, 'Reshma': 89}
Descending order: ['Anu', 'Reshma', 'Vismaya', 'Neema', 'Swathi', 'Riya']
Ascending order: ['Riya', 'Swathi', 'Neema', 'Vismaya', 'Reshma', 'Anu']
>>>
```

PROGRAM NO: 17

AIM: Merge two dictionaries

ALGORITHM:

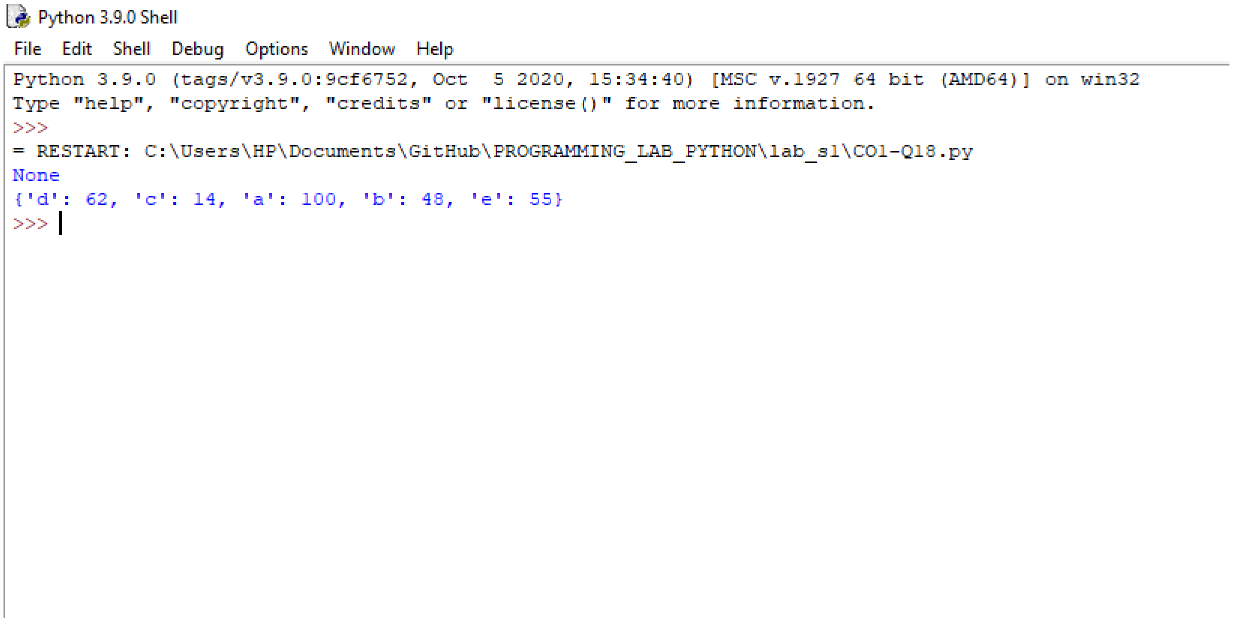
```
Step1:Start
Step2: Define a function Merge with 2 arguments dict1 and dict 2 as two dictionaries
Step3: Give key :value pairs to both dicts
Step4: Merged two dicts using Merge ()function
Step 5: Displayed the resultant dict as in dict2
Step6:Stop
```

PROGRAM:

```
def Merge(dict1, dict2):
    return (dict2.update(dict1))

dict1 = {'a': 100, 'b': 48, 'e': 55}
dict2 = {'d': 62, 'c': 14}
print(Merge(dict1, dict2))
print(dict2)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q18.py
None
{'d': 62, 'c': 14, 'a': 100, 'b': 48, 'e': 55}
>>> 
```

PROGRAM NO: 18

AIM: Find gcd of 2 numbers.

ALGORITHM:

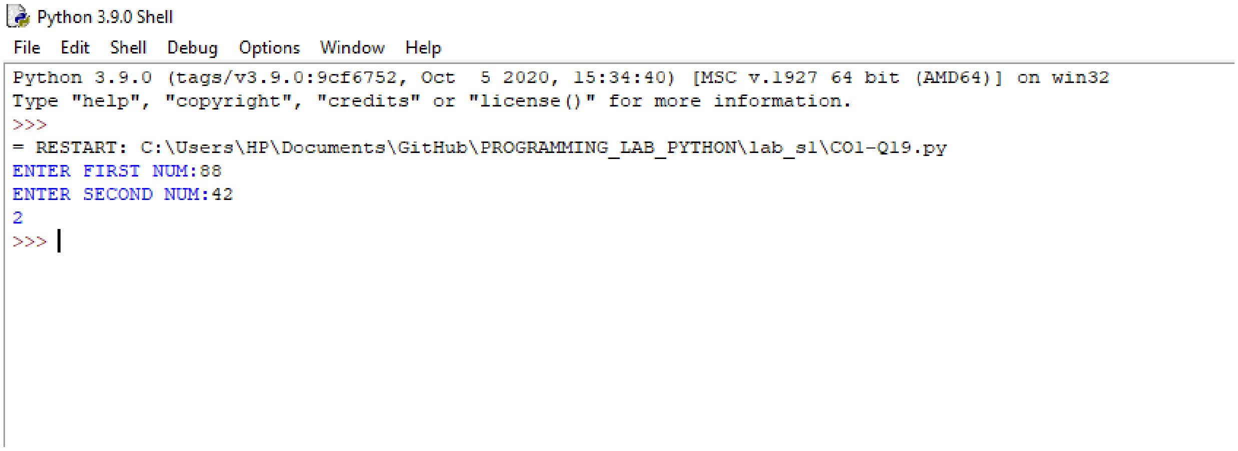
```
Step1: Start
 Step2: Take input as two numbers n1 and n2
Step2. Take gcd=1
Step3. Perform the following step till k in range n2/2 and 0
Check if ((x % k == 0) and (y % k== 0)), then assign GCD=k
Step4. Print the value of gcd
Step5:Stop
```

PROGRAM:

```
n1=int(input("ENTER FIRST NUM:"))
n2=int(input("ENTER SECOND NUM:"))
gcd=1
if n1%n2==0:
    print(n2)
for k in range(int(n2 / 2), 0, -1):
        if n1 % k == 0 and n2 % k == 0:
            gcd = k
            break
print(k)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q19.py
ENTER FIRST NUM:88
ENTER SECOND NUM:42
2
>>> 
```

PROGRAM NO: 19

AIM: From a list of integers, create a list removing even numbers.

ALGORITHM:

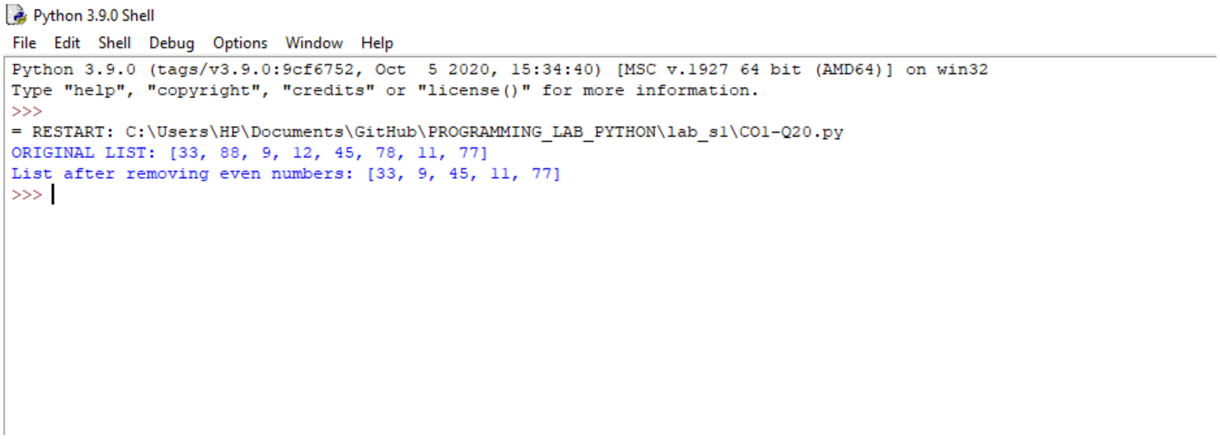
```
Step1: Start
Step2: Take a list of numbers as input
Step3: For each value in list, check for even number
Step4: If found,then remove it from list by using remove()
Step5: Then display the resultant list as output
Step6: Stop
```

PROGRAM:

```
li=[33,88,9,12,45,78,11,77]
print("ORIGINAL LIST:",li)
for i in li:
    if (i % 2 ==0):
        li.remove(i)

print("List after removing even numbers:",li)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO1-Q20.py
ORIGINAL LIST: [33, 88, 9, 12, 45, 78, 11, 77]
List after removing even numbers: [33, 9, 45, 11, 77]
>>> 
```

COURSE OUTCOME 2

PROGRAM NO: 20

AIM: Program to find the factorial of a number

ALGORITHM:

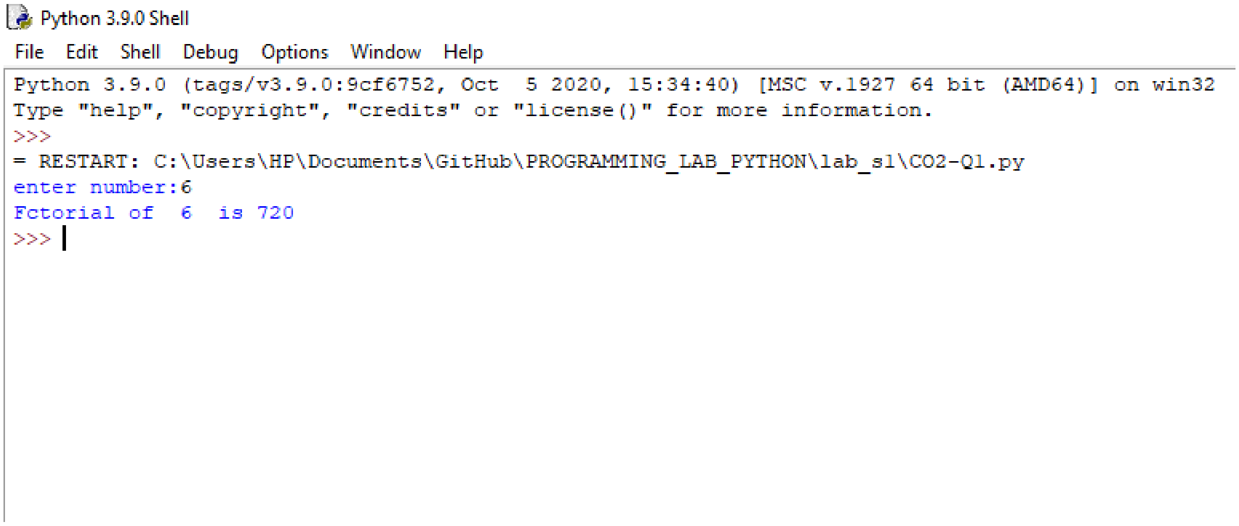
```
Step1: Start
Step2: Read n and initialize fact-=0
Step3: if n<0 then cannot find factorial
Step4: Elseif n==0 then factorial is 0
Step5: Else find factorial in range 1 to n+1
Step6: Display the factorial of n as output
Step7: Stop
```

PROGRAM:

```
n=int(input("enter number:"))
fact=1
if n<0:
    print("cannot find factorial")
elif n==0:
    print("Factorial is 0")
else:
    for i in range(1,n+1):
        fact=fact*i
print("Fctorial of ",n," is",fact)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-Q1.py
enter number:6
Fctorial of  6  is 720
>>> 
```

PROGRAM NO: 21

AIM: Generate Fibonacci series of N terms

ALGORITHM:

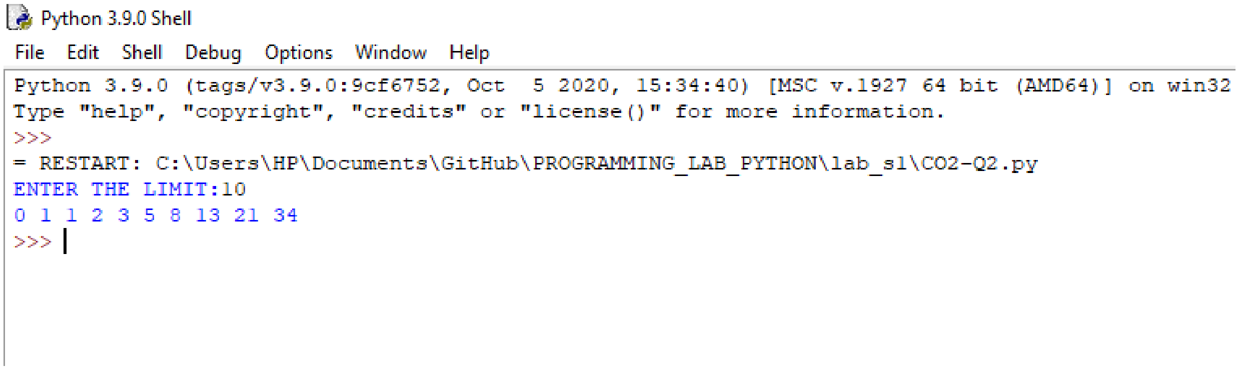
```
Step1: Start
Step2: Take limit value as input n
Step3: Set f =0 and s=1 then, check whether n <=0
Step4: Then Fibonacci series is f
    Else print the series from range 2 to the limit
Step5: Display the result as output
Step6: Stop
```

PROGRAM:

```
n=int(input("ENTER THE LIMIT:"))
f=0
s=1
if n<=0:
    print("The requested series is",f)
else:
    print(f,s,end=" ")
    for x in range(2,n):
        next=f+s
        print(next,end=" ")
        f=s
        s=next
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-Q2.py
ENTER THE LIMIT:10
0 1 1 2 3 5 8 13 21 34
>>> 
```

PROGRAM NO: 22

AIM: Find the sum of all items in a list

ALGORITHM:

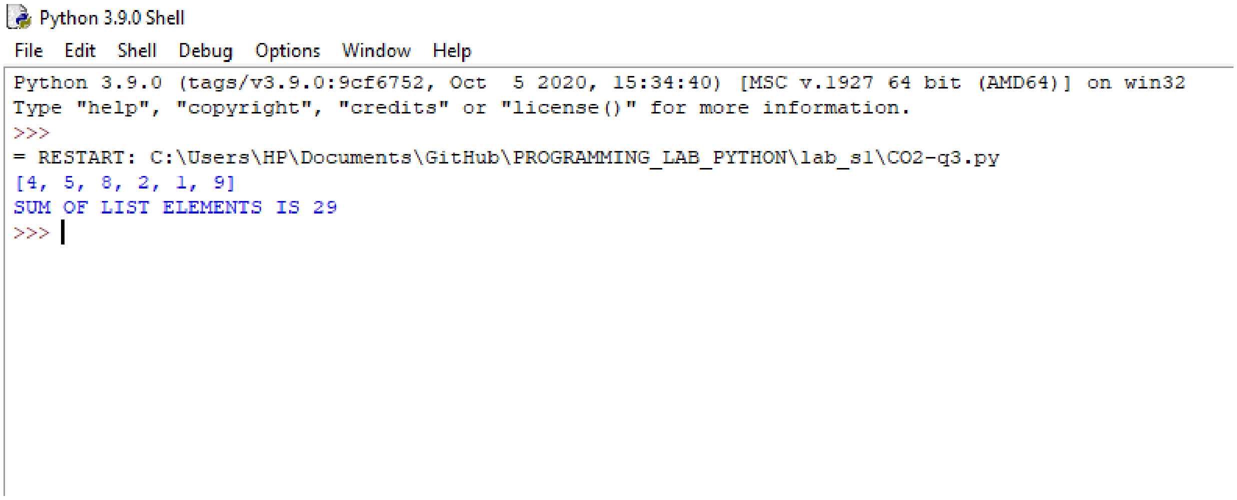
```
Step1: Start
Step2: Take a list of numbers as input
Step3: Find the sum of values in list using sum()
Step4: Display the sum result as output
Step5: Stop
```

PROGRAM:

```
a=[4,5,8,2,1,9]
print(a)
b=sum(a)
print("SUM OF LIST ELEMENTS IS",b)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q3.py
[4, 5, 8, 2, 1, 9]
SUM OF LIST ELEMENTS IS 29
>>> 
```

PROGRAM NO: 23

AIM: Generate a list of four digit numbers in a given range with all their digits even andthe number is a perfect square.

ALGORITHM:

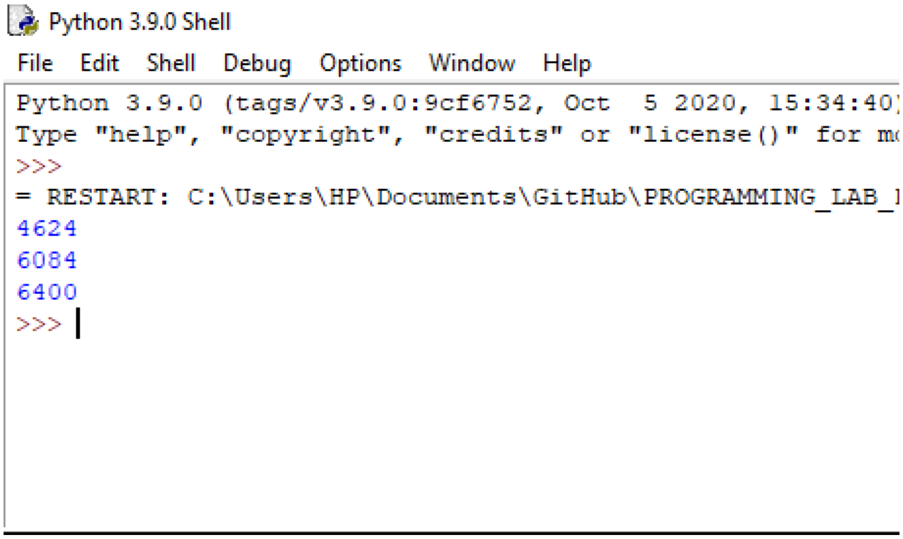
Step1: Start
Step2: Take range from 1000 to 8000
Step3: Then check for square roots and even numbers between that range
Step4: And display the result
Step5: Stop

PROGRAM

```
      import math
for i in range(1000,8000):
  n=int(math.sqrt(i))
   if n*n==i:
    s=i
    while s!=0:
    r=s%10
    s=s//10
    if (r%2!=0):
        break
else:
     print(i)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40)
Type "help", "copyright", "credits" or "license()" for mo
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_I
4624
6084
6400
>>>
```

PROGRAM NO: 24

AIM: Display the given pyramid with step number accepted from user.

Eg: N=4

```
1

2 4

3 6 9

4 8 12 16
```

ALGORITHM:

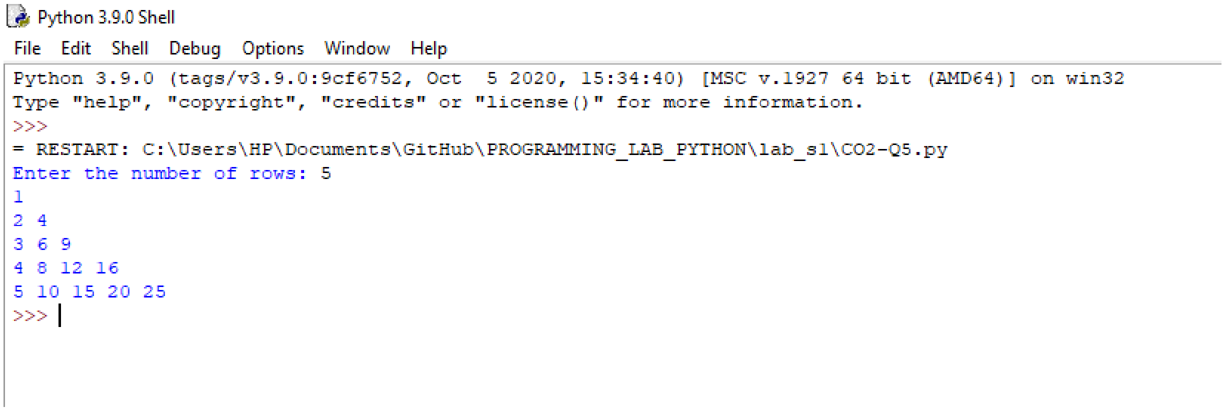
```
Step1: Start
Step2: Take number of rows as input
Step3: For i value in range 1 to row number and j value in range 1 to i +1
Step4: Display i*j as pattern
Step5: Stop
```

PROGRAM:

```
rows =int(input("Enter the number of rows: "))
for i in range(1,rows+1):
    for j in range( 1,i+1 ):
        print(i*j, end=' ')

    print(" ")
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-Q5.py
Enter the number of rows: 5
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
>>> |
```

PROGRAM NO: 25

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

ALGORITHM:

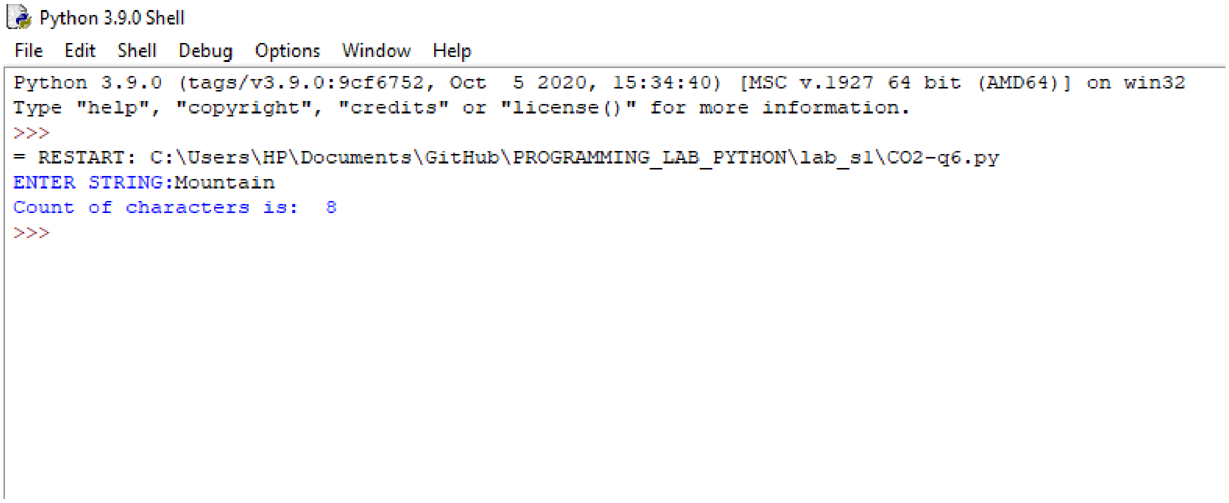
```
Step1: Start
Step2: Take a string as input
Step3: For each character in the string,the count variable gets incremented
Step4: Then finally display the value of count as character frequency
Step5: Stop
```

PROGRAM:

```
st=str(input("ENTER STRING:"))
count=0
for i in st:
    count=count+1
print("Count of characters is: " ,count)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q6.py
ENTER STRING:Mountain
Count of characters is:  8
>>> 
```

PROGRAM NO: 26

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

ALGORITHM:

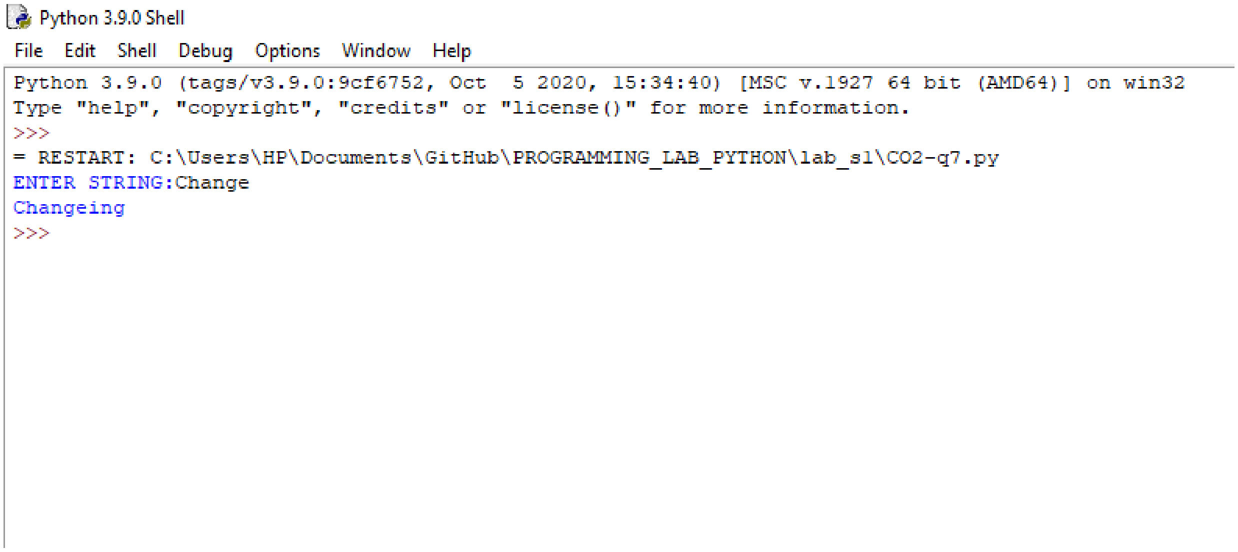
```
Step1: Start
Step2: Take an input string ‘s’
Step3: Then check for ‘ing’ at the 3rd last position
Step4: If found replace that ‘ing’ with ‘ly’
Step5: Display resultant string
Step6: Stop
```

PROGRAM:

```
s=str(input("ENTER STRING:"))
if s[-3:] == "ing":
    s += "ly"

    print(s)
else:
    s+="ing"
    print(s)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q7.py
ENTER STRING:Change
Changeing
>>> 
```

PROGRAM NO: 27

AIM: Accept a list of words and return length of longest word.

ALGORITHM:

Step1: Start

Step2: Take input as list of words separated by ‘,’

Step3: Split each word and define len_log() function to find length of longest word

Step4: Then place that longest word’s length as max and display it as result

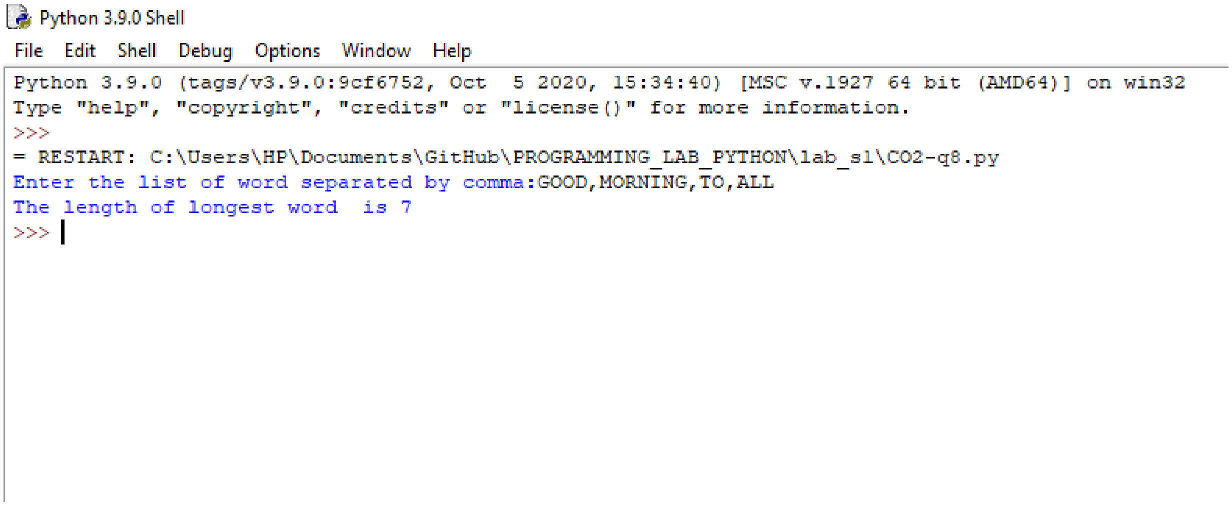
Step5:Stop

PROGRAM:

```
lst=input("Enter the list of word separated by comma:")


a = lst.split(",")
def len_log(list1):
    max=len(list1[0])
    for i in list1:
        if len(i)>max:
            max=len(i)
    return  max
print("The length of longest word  is",len_log(a))
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q8.py
Enter the list of word separated by comma:GOOD,MORNING,TO,ALL
The length of longest word  is 7
>>> 
```

PROGRAM NO:28

AIM: Construct following pattern using nested loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

ALGORITHM:

Step1:Start

Step2:Take number of rows as input ‘rows’

Step3:create the given pattern inrange from 0 to rows

Step4:Then reverse the for condition for creating second pattern

Step5:Then display the pattern’*’

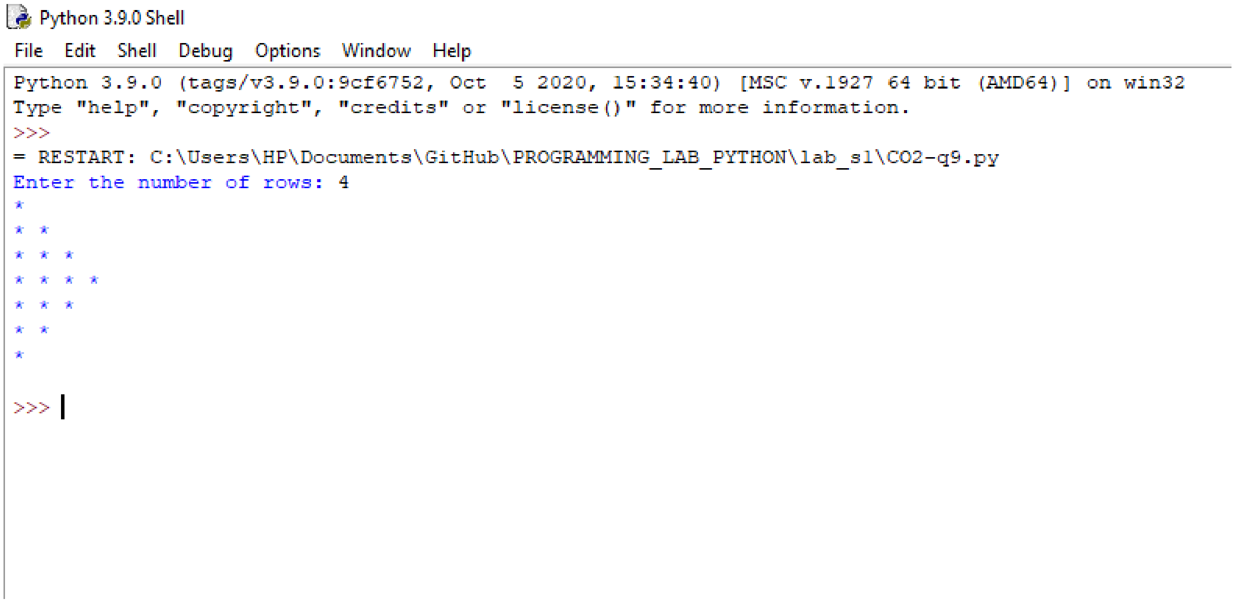
Step6:Stop

PROGRAM:

```
rows = int(input("Enter the number of rows: "))
for i in range(0, rows):
    for j in range(0, i + 1):
        print("*", end=' ')
    print(" ")
# For second pattern
for i in range(rows , 0, -1):
    for j in range(0, i - 1):
        print("*", end=' ')
    print(" ")
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q9.py
Enter the number of rows: 4
*
* *
* * *
* * * *
* * *
* *
*

>>> 
```

PROGRAM NO: 29

AIM: Generate all factors of a number.

ALGORITHM:

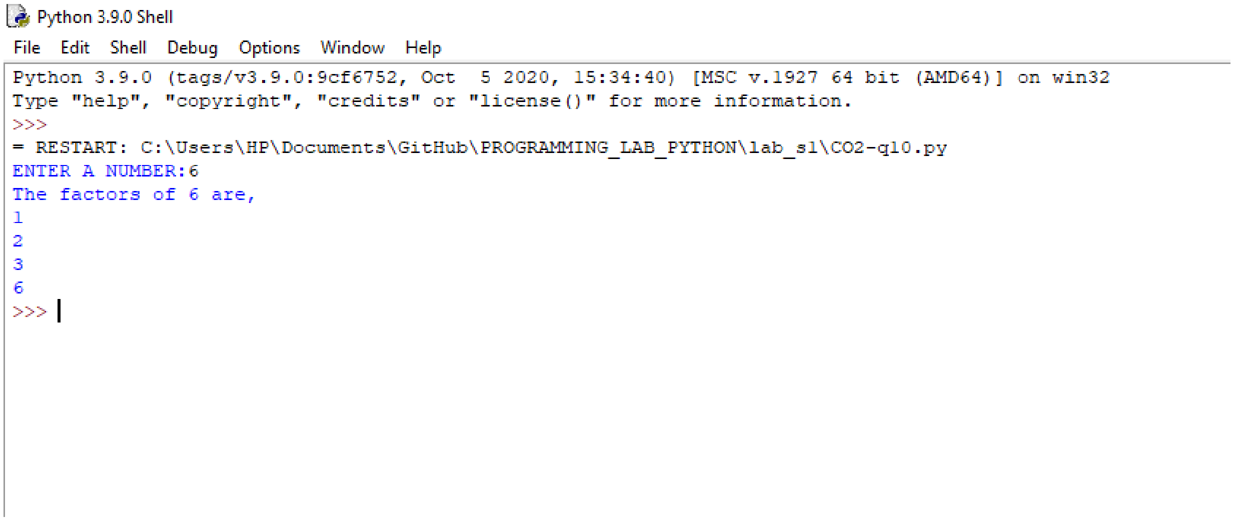
```
Step1:Start
Step2:Take N as input number
Step3:Check for i in range 1 to N+1
      If N% i ==0,then print i
Step3:Display that i values as factors of N
Step4:Stop
```

PROGRAM:

```
N=int(input("ENTER A NUMBER:"))
print("The factors of {} are,".format(N))
for i in range(1,N+1):
    if N % i == 0:
        print(i)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q10.py
ENTER A NUMBER:6
The factors of 6 are,
1
2
3
6
>>> 
```

PROGRAM NO: 30

AIM: Write lambda functions to find area of square, rectangle and triangle

ALGORITHM:

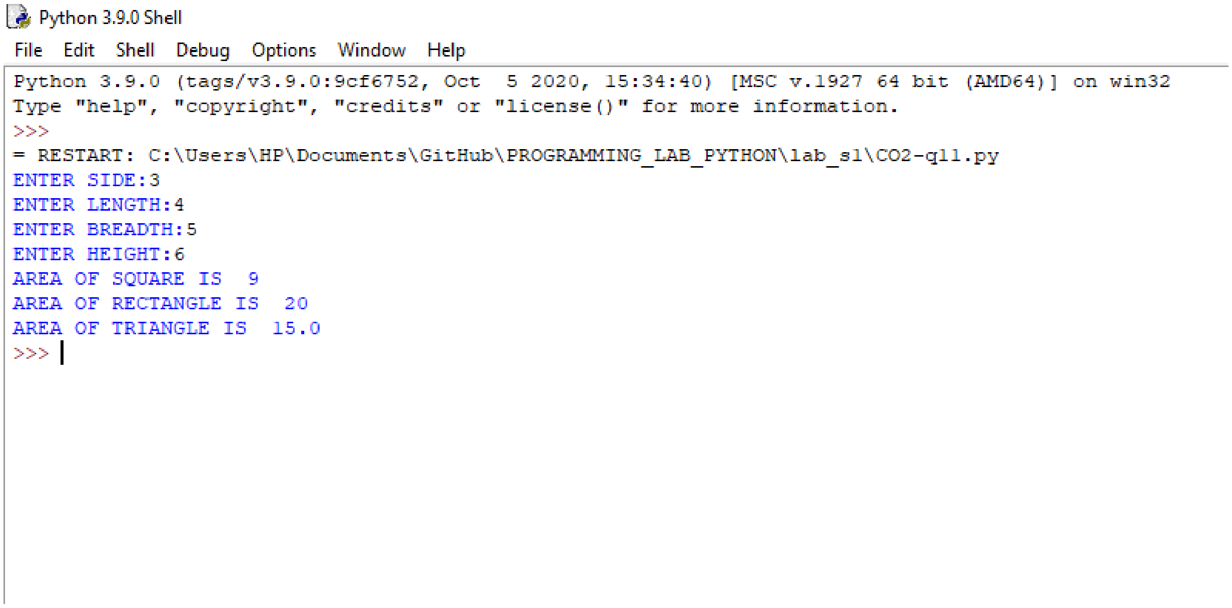
```
Step1: Start
Step2: Take inputs as S for side,l for length, b for breadth ,h for height
Step3: Use three lambda functions as x,y,z
          x for area of square
          y for area of rectangle
         z for area of triangle
Step4: Find the areas in x, y, z
Step5: Display the results of x, y , z
Step6:Stop
```

PROGRAM:

```
S=int(input("ENTER SIDE:"))
l=int(input("ENTER LENGTH:"))
b=int(input("ENTER BREADTH:"))
h=int(input("ENTER HEIGHT:"))
x = lambda a : a * a
print("AREA OF SQUARE IS ",(x(S)))
y = lambda a, b : a * b
print("AREA OF RECTANGLE IS ",(y(l, b)))
z = lambda a, b : 1/2 * (a * b)
print("AREA OF TRIANGLE IS ",(z(b, h)))
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO2-q11.py
ENTER SIDE:3
ENTER LENGTH:4
ENTER BREADTH:5
ENTER HEIGHT:6
AREA OF SQUARE IS  9
AREA OF RECTANGLE IS  20
AREA OF TRIANGLE IS  15.0
>>> 
```

COURSE OUTCOME 3

PROGRAM NO: 31

AIM: Work with built-in packages

ALGORITHM:

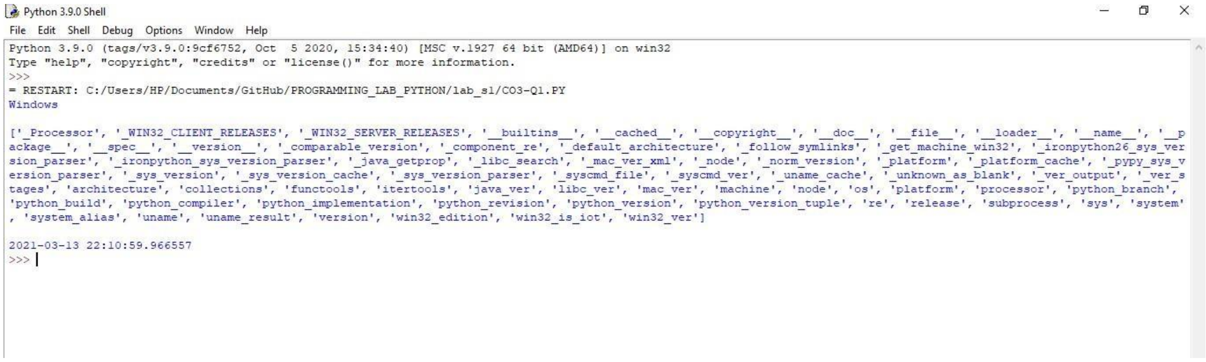
```
Step1: Start
Step2: Imported platform , datetime built-in packages using import keyword
Step3: Then use functions in platform like system(),dir(platform) and datetime.now() in datetime module
Step4: The results of each function is displayed as output
Step5: Stop
```

PROGRAM:

```
import platform
x =platform.system()
print(x)
print(end="\n")
import platform
x = dir(platform)
print(x)
print(end="\n")
import datetime
x = datetime.datetime.now()
print(x)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/HP/Documents/GitHub/PROGRAMMING_LAB_PYTHON/lab_s1/CO3-Q1.PY
Windows

['_Processor', '_WIN32_CLIENT_RELEASES', '_WIN32_SERVER_RELEASES', '__builtins__', '__cached__', '__copyright__', '__doc__', '__file__', '__loader__', '__name__', '__p
ackage__', '__spec__', '__version__', '_comparable_version', '_component_re', '_default_architecture', '_follow_symlinks', '_get_machine_win32', '_ironpython26_sys_ver
sion_parser', '_ironpython_sys_version_parser', '_java_getprop', '_libc_search', '_mac_ver_xml', '_node', '_norm_version', '_platform', '_platform_cache', '_pypy_sys_v
ersion_parser', '_sys_version', '_sys_version_cache', '_sys_version_parser', '_syscmd_file', '_syscmd_ver', '_uname_cache', '_unknown_as_blank', '_ver_output', '_ver_s
tages', 'architecture', 'collections', 'functools', 'itertools', 'java_ver', 'libc_ver', 'mac_ver', 'machine', 'node', 'os', 'platform', 'processor', 'python_branch',
'python_build', 'python_compiler', 'python_implementation', 'python_revision', 'python_version', 'python_version_tuple', 're', 'release', 'subprocess', 'sys', 'system'
, 'system_alias', 'uname', 'uname_result', 'version', 'win32_edition', 'win32_is_iot', 'win32_ver']

2021-03-13 22:10:59.966557
>>> 
```

PROGRAM NO: 32

AIM: 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)

ALGORITHM:

Step1: Start

Step2: Created a package named ‘Graphics’ with modules ‘Circle.py’ and ‘Rectangle.py’

Step3: Inside this Graphics package, create a sub-package called ‘3D-graphics’ with modules ‘Cuboid.py’ and ‘Sphere.py’

Step4: Each module is created with respective methods for finding area and perimeter of all figures

Step5: Then by using selective import and import *, the results of area and perimeter are displayed

Step6: Stop

PROGRAM:

Graphics

- Graphics3D
  - __init .py
  - Cuboid.py
  - Sphere.py
- __init .py
- Circle.py
- Rectangle.py

pkg.py

Circle.py

```
from math import pi
def area(r):
    return pi*r*r
def perimeter(r):
    return 2*pi*r
```

Rectangle.py

```
def Rarea(l,b):
    return l*b
def Rperimeter(l,b):
    return  (2*(l+b))
```

Cuboid.py

```
def CUarea(l,w,h):
    return 2*(l*w+l*h+h*w)
def CUperimeter(l,w,h):
    return 4*(l+w+h)
```

Sphere.py

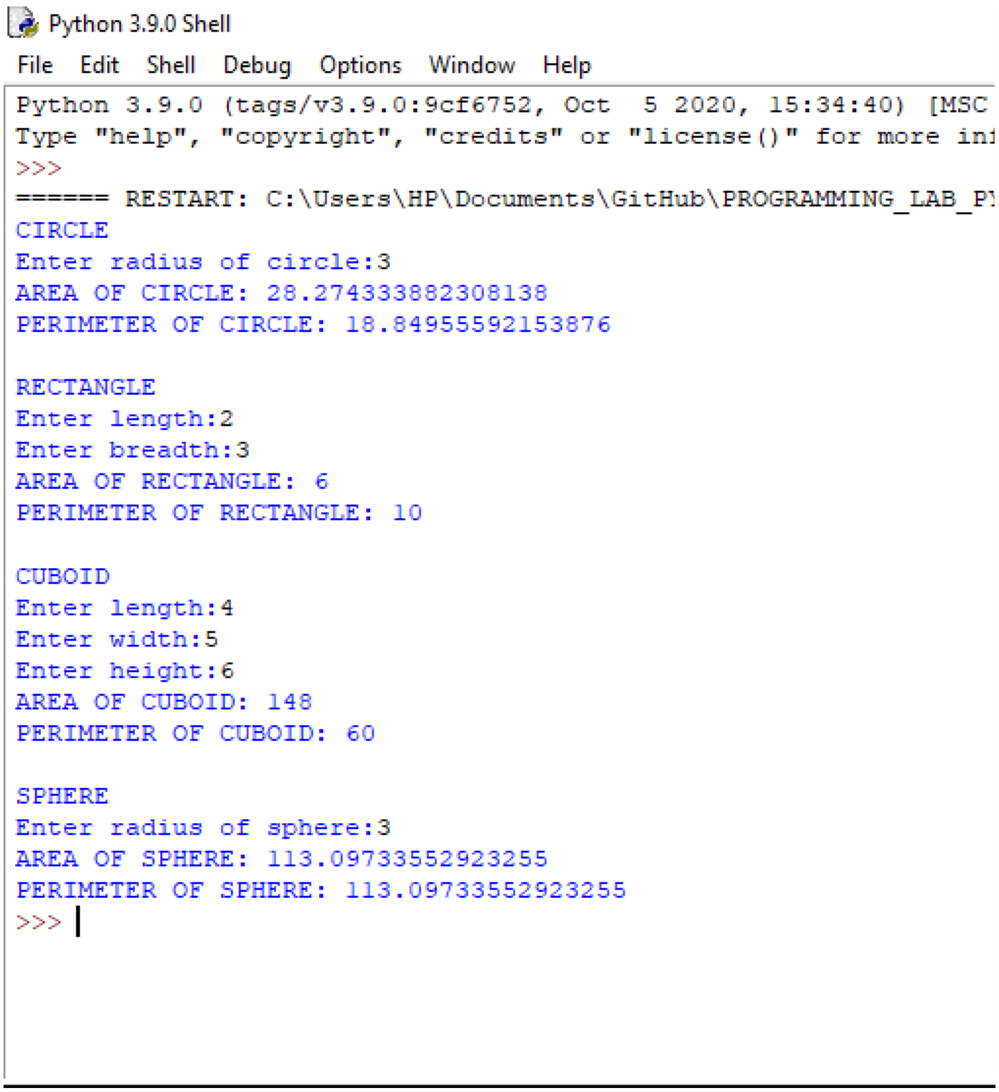
```
from math import pi
def Sarea(r):
    return 4*pi*r*r
def Sperimeter(r):
    return ((4/3)*pi*r*r*r)
```

pkg.py

```
from Graphics.Circle import *

from Graphics.Rectangle import *

from Graphics.Graphics3D.Cuboid import *

from Graphics.Graphics3D.Sphere import *


print("CIRCLE")

r=int(input("Enter radius of circle:"))

print("AREA OF CIRCLE:",area(r))
print("PERIMETER OF CIRCLE:",perimeter(r))

print("\nRECTANGLE")
l=int(input("Enter length:"))
b=int(input("Enter breadth:"))
print("AREA OF RECTANGLE:",Rarea(l,b))
print("PERIMETER OF
RECTANGLE:”,Rperimeter(l,b))

print("\nCUBOID")
l=int(input("Enter length:"))
w=int(input("Enter width:"))
h=int(input("Enter height:"))
print("AREA OF CUBOID:",CUarea(l,w,h))
print("PERIMETER OF
CUBOID:",CUperimeter(l,w,h))

print("\nSPHERE")
r=int(input("Enter radius of sphere:"))
print("AREA OF SPHERE:",Sarea(r))
print("PERIMETER OF SPHERE:",Sperimeter(r))
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC
Type "help", "copyright", "credits" or "license()" for more inf
>>>
====== RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PY
CIRCLE
Enter radius of circle:3
AREA OF CIRCLE: 28.274333882308138
PERIMETER OF CIRCLE: 18.84955592153876

RECTANGLE
Enter length:2
Enter breadth:3
AREA OF RECTANGLE: 6
PERIMETER OF RECTANGLE: 10

CUBOID
Enter length:4
Enter width:5
Enter height:6
AREA OF CUBOID: 148
PERIMETER OF CUBOID: 60

SPHERE
Enter radius of sphere:3
AREA OF SPHERE: 113.09733552923255
PERIMETER OF SPHERE: 113.09733552923255
>>> |
```

COURSE OUTCOME 4

PROGRAM NO: 33

AIM: Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.

ALGORITHM:

```
Step1: Start
Step2: Created a class ‘Rectangle’ with attributes length and breadth
Step3: In that class ,create 2 methods to find area and perimeter of rectangle object
Step4: Then created two rectangle object R1 andR2 to compare their areas by using the defined methods in ‘Rectangle’ class
Step5: Stop
```

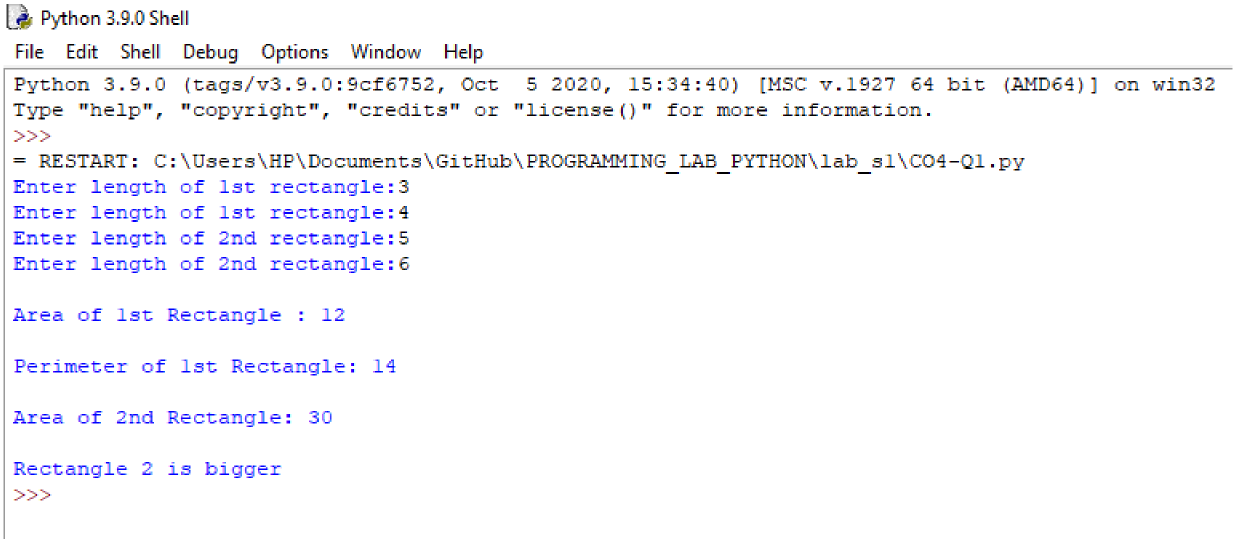
PROGRAM:

```
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 compare(R1,R2):
     if R1.area()>R2.area():
         print("\n Rectangles 1 is bigger")
     else:
         print("\nRectangle 2 is bigger")

     return

a=int(input("Enter length of 1st rectangle:"))
b=int(input("Enter length of 1st rectangle:"))
c=int(input("Enter length of 2nd rectangle:"))
d=int(input("Enter length of 2nd rectangle:"))
R1=Rectangle(a,b)
R2=Rectangle(c,d)
print("\nArea of 1st Rectangle :",R1.area())
print("\nPerimeter of 1st Rectangle:",R1.perimeter())
print("\nArea of 2nd Rectangle:",R2.area())
R1.compare(R2)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO4-Q1.py
Enter length of 1st rectangle:3
Enter length of 1st rectangle:4
Enter length of 2nd rectangle:5
Enter length of 2nd rectangle:6

Area of 1st Rectangle : 12

Perimeter of 1st Rectangle: 14

Area of 2nd Rectangle: 30

Rectangle 2 is bigger
>>> 
```

PROGRAM NO: 34

AIM: 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.

ALGORITHM:

Step1: Start

Step2: Create a class named ‘Bankaccount’

Step2: Defined an_init_method with attributes accno, name, typeofacc ,balance as taken as inputs

Step3: Then defined functions like deposit (),withdraw() and display() to display the amount deposited; amount withdrawed and to display available balance

Step4: Then created an object ‘s’ of class ‘Bankaccount’ to display all the above functions results

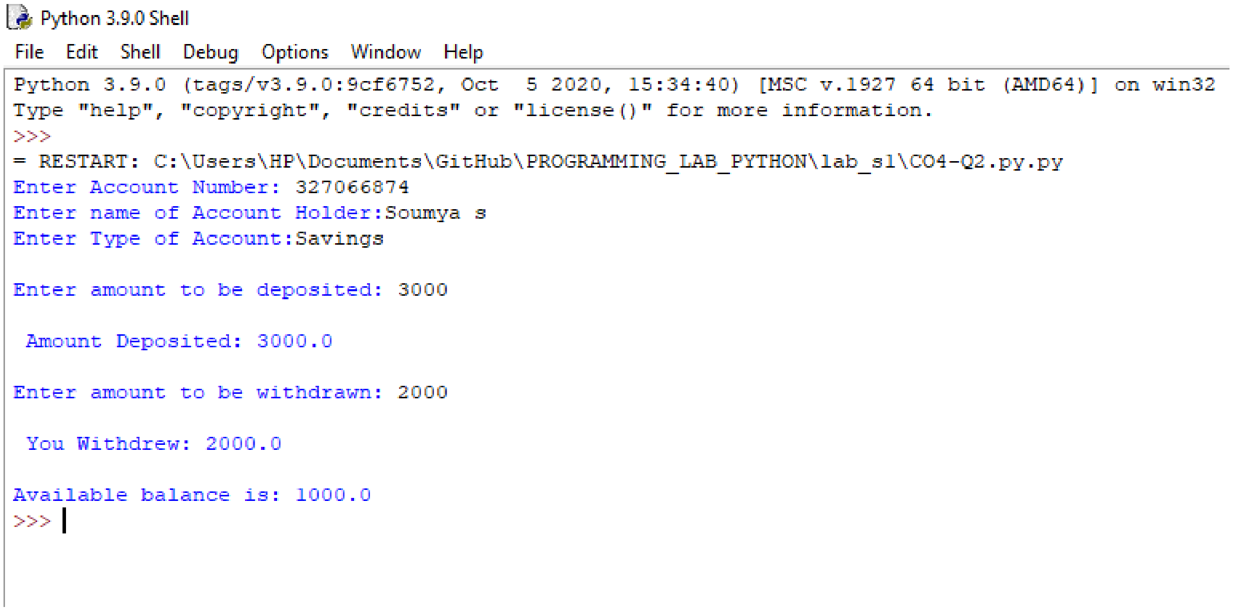
Step5: Stop

PROGRAM:

```
class Bankaccount:
    def __init__(self):
        accno=float(input("Enter Account Number: "))
        name=str(input("Enter name of Account Holder:"))
        typeofacc=str(input("Enter Type of Account:"))
        self.balance=0
    def deposit(self):
        amount = float(input("\nEnter amount to be deposited: "))
        self.balance += amount
        print("\n Amount Deposited:", amount)
    def withdraw(self):
        amount = float(input("\nEnter amount to be withdrawn: "))
        if self.balance >= amount:
            self.balance -= amount
            print("\n You Withdrew:", amount)
        else:
            print("\n Insufficient balance ")

    def display(self):
        print("\nAvailable balance is:",self.balance)

s = Bankaccount()
s.deposit()
s.withdraw()
s.display()
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO4-Q2.py.py
Enter Account Number: 327066874
Enter name of Account Holder:Soumya s
Enter Type of Account:Savings

Enter amount to be deposited: 3000

 Amount Deposited: 3000.0

Enter amount to be withdrawn: 2000

 You Withdrew: 2000.0

Available balance is: 1000.0
>>> 
```

PROGRAM NO: 35

AIM: Create a class Rectangle with private attributes length and width. Overload ‘<’ operator to compare the area of 2 rectangles.

ALGORITHM:

Step1: Start

Step2: Firstly create a class named ‘Rectangle’ with private attributes length and width

Step3: Then define a function area() to find the area of rectangle

Step4: Then defined another method__lt__to compare the areas of 2 rectangle

Step5: By using two objects obj1 and obj 2 ,compared the areas with overloading ‘<’ operator

Step6: Stop

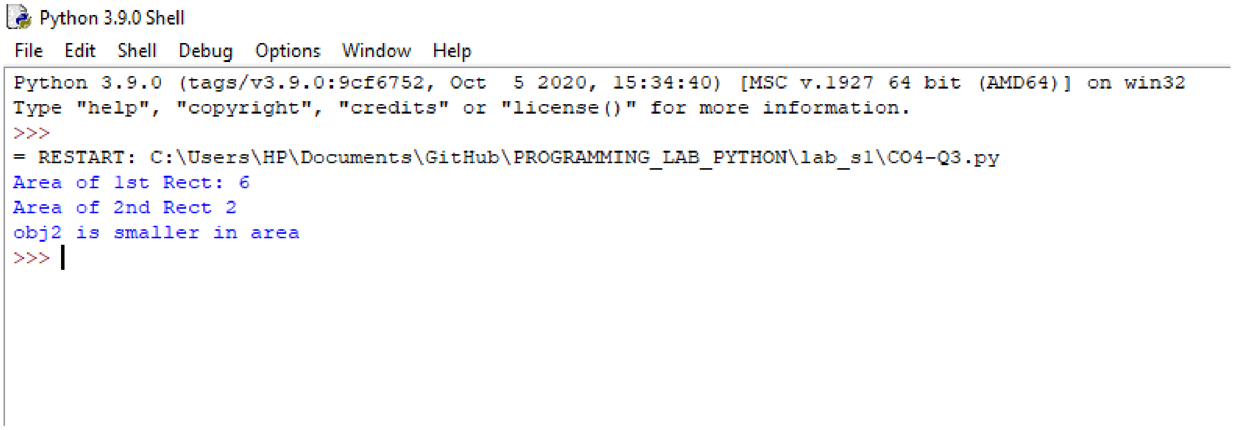
PROGRAM:

```
class Rectangle:
    __length=0
    __width=0
    __area=0
    def_init_(self,l,w):
        self._ length=l;
        self._width=w;
    def area(self):
        return  self._ length*self._ width
    def_lt_(self,other):
        if(self._area<other._area):
            return True
        else:
            return False
```

```
obj1=Rectangle(2,3)
obj2=Rectangle(1,2)
print("Area of 1st Rect:",obj1.area())
print("Area of 2nd Rect",obj2.area())
if(obj1.area()<obj2.area()):
    print("obj1 is smaller in area")
else:
    print("obj2 is smaller in area")
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO4-Q3.py
Area of 1st Rect: 6
Area of 2nd Rect 2
obj2 is smaller in area
>>> 
```

PROGRAM NO: 36

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

ALGORITHM:

Step1: Start

Step2: Created a class 'Time' with private attributes hour,minute,second

Step3: Then defined a function 'time(self)' and give the condition for incrementing hour ,minute when the value is greater than 60

Step4: Then created another function 'add()' to add two times

Step5: Then use + operator to overload and add the two inputed times

Step6: Displays the sum as output

Step7: Stop

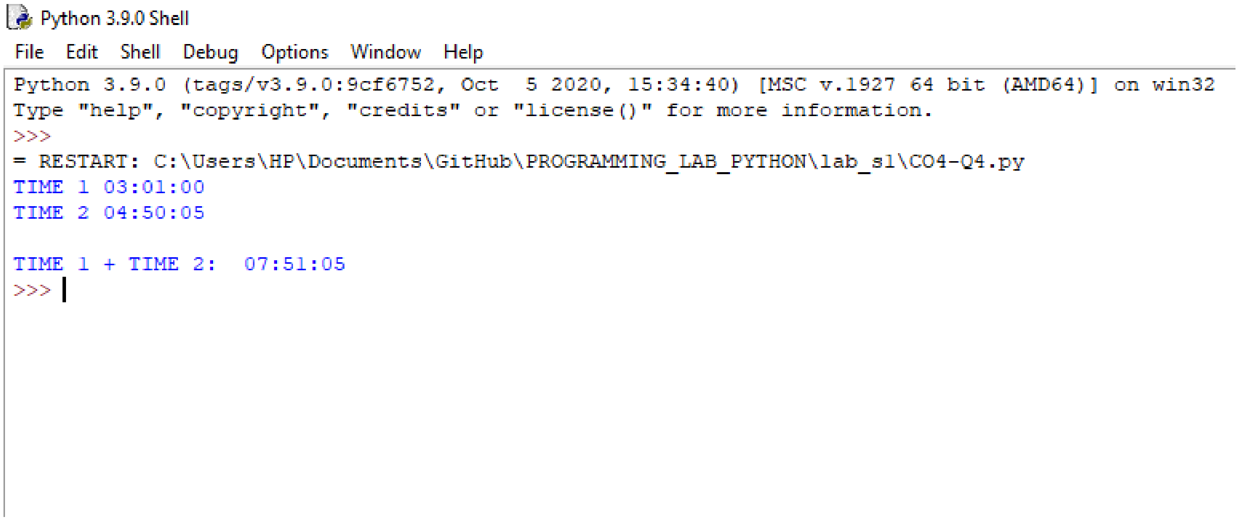
PROGRAM

```
class Time:
   def_init_(self,h,m,s):
      self._hour=h
      self._minute=m
      self._second=s

   def time(self):
      if self._second>=60:
         self._second-=60
         self._minute+=1
```

```
        if self._minute>=60:
            self._ minute-=60
            self._hour+=1
        return("%.2d:%.2d:%.2d"%(self._hour,self._ minute,self._second))
    def_add_(self,other):
        __hour=self._hour+other._hour
        __minute=self._minute+other._minute
        __second=self._second+other._second
        return("%.2d:%.2d:%.2d"%(_hour,_minute,_second))


t1=Time(2,60,60)
print("TIME 1",t1.time())
t2=Time(4,50,5)
print("TIME 2",t2.time())
print("\nTIME 1 + TIME 2: ",(t1 + t2))
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO4-Q4.py
TIME 1 03:01:00
TIME 2 04:50:05

TIME 1 + TIME 2:  07:51:05
>>> 
```

PROGRAM NO: 37

AIM: 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.

ALGORITHM:

Step1: Start

Step2: Create a class named ‘Publisher’ with attribute ‘Pubname’

Step3: Derived a class Book from Publisher with attribute ‘title’ and ‘author’

Step4: Then derived a class ‘Python’ from the class ‘Book’

Step5: Then use base class constructor

Step6: Also used method overriding to override multiple methods from different classes to display the information about book ‘Wings of Fire’

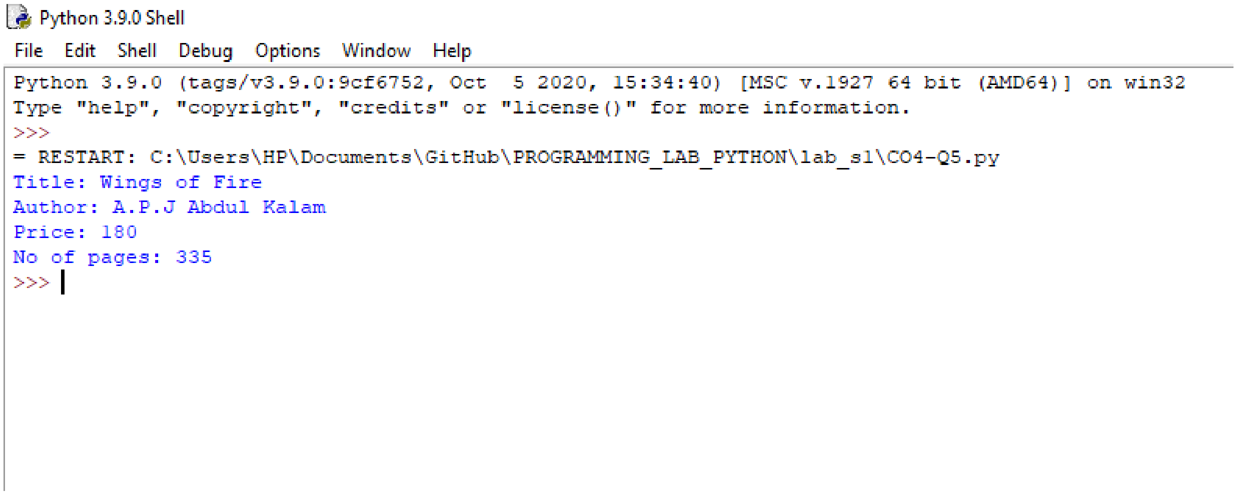
Step7: Stop

PROGRAM

```
class Publisher:
    def_init_(self,Pubname):
        self.Pubname=Pubname
    def display(self):
        print("Publisher name is:",self.Pubname)



class Book(Publisher):
    def_init_(self,Pubname,title,author):
        Publisher._init_(self,Pubname)
        self.title=title
        self.author=author
    def display(self):
        print("Title:",self.title)
        print("Author:",self.author)


class Python(Book):
    def_init_(self,Pubname,title,author,price,no_of_pages):
        Book._init_(self,Pubname,title,author)
        self.price=price
        self.no_of_pages=no_of_pages
    def display(self):
        print("Title:",self.title)
        print("Author:",self.author)
        print("Price:",self.price)
        print("No of pages:",self.no_of_pages)
b1=Python("DC BOOKS","Wings of Fire","A.P.J Abdul Kalam", 180, 335)
b1.display()
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\CO4-Q5.py
Title: Wings of Fire
Author: A.P.J Abdul Kalam
Price: 180
No of pages: 335
>>> 
```

COURSE OUTCOME 5

PROGRAM NO : 38

AIM: Write a Python program to read a file line by line and store it into a list.

ALGORITHM:

Step1: Start

Step2: Open a file named Afile.txt and write the content in string str1

Step3: Then read the files line by line and store it in str

Step4: Then display each line in st2 in list lst

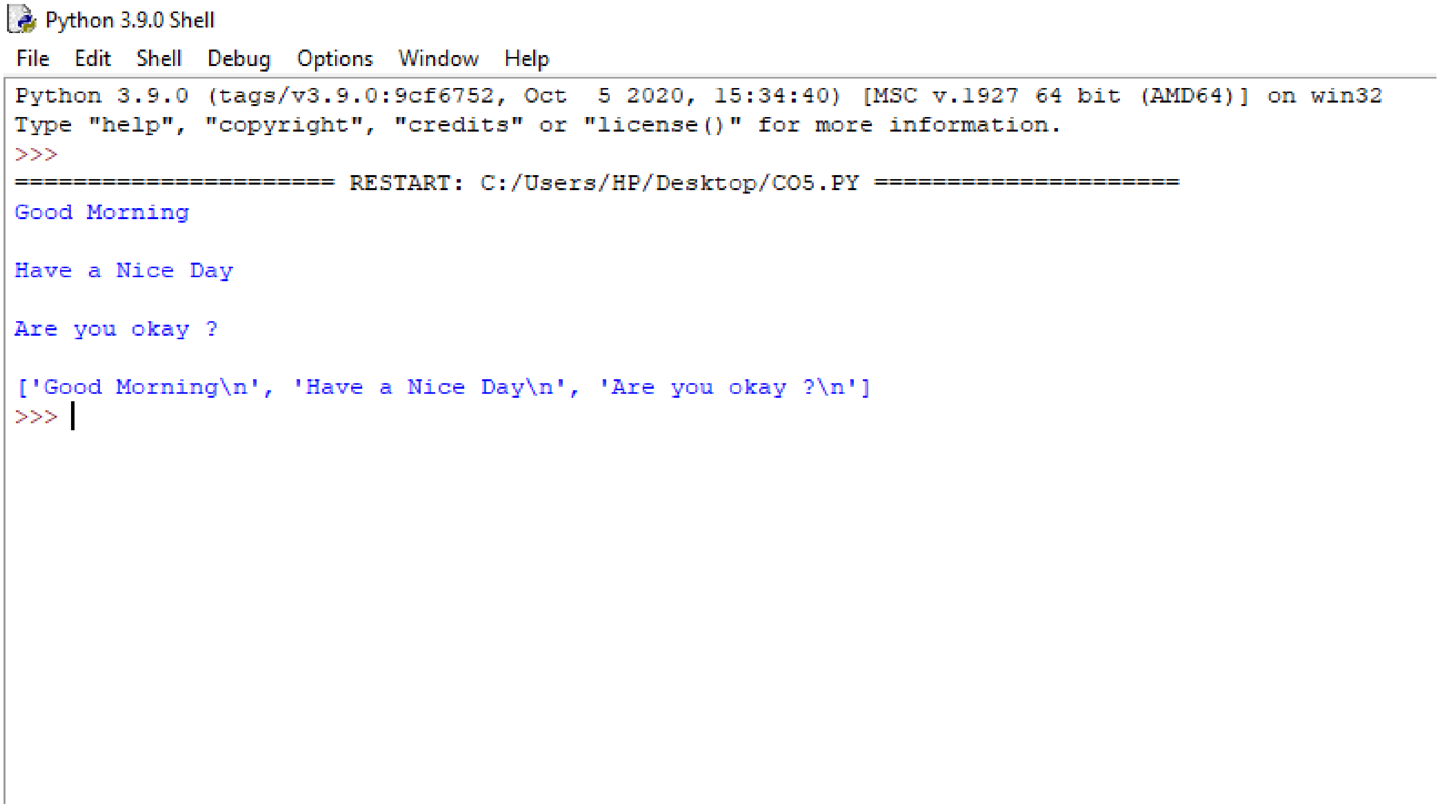
Step5: Stop

PROGRAM:

```
  st1="Good Morning""\n""Have a Nice Day""\n""Are you okay ?""\n"
  fw=open("Afile.txt","w")
 fw.write(st1)
 fw.close()

  fr=open("Afile.txt","r")
   st2=fr.readlines()
lst=[]
for i in st2:
 print(i)
lst.append(i)
print(lst)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
====================== RESTART: C:/Users/HP/Desktop/CO5.PY =====================
Good Morning

Have a Nice Day

Are you okay ?

['Good Morning\n', 'Have a Nice Day\n', 'Are you okay ?\n']
>>> 
```

PROGRAM NO : 39

AIM: Python program to copy odd lines of one file toother

ALGORITHM:

Step1: Start

Step2: Create two files named “sample.txt “and “sample2.txt”

Step3: Write some lines of data into “sample.txt”

Step4: Then copy the odd lines of “ sample.txt” into “sample2.txt” and display it as output

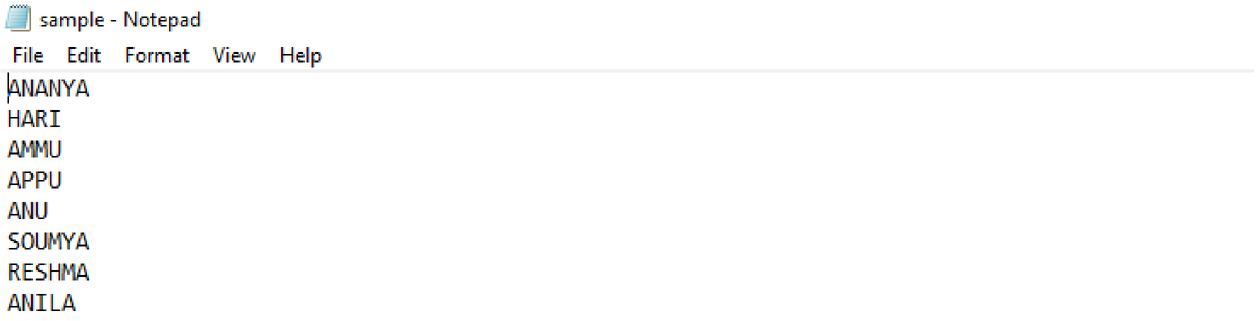
Step5: Stop

PROGRAM:

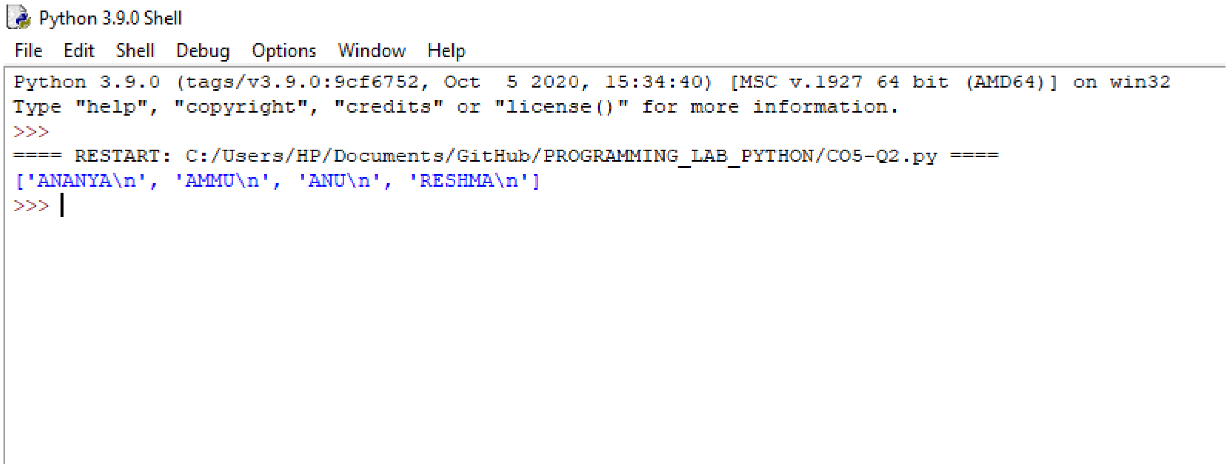
```
f = open("sample.txt",'r')
str1 = f.readlines()
f.close()
f = open("sample2.txt",'w')
x = 0;
for i in str1:
    x = x+1
    if x%2!=0:
        f.write(i)
f.close()
f=open("sample2.txt",'r')
str2=f.readlines()
print(str2)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
ANANYA
HARI
AMMU
APPU
ANU
SOUMYA
RESHMA
ANILA
```



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
==== RESTART: C:/Users/HP/Documents/GitHub/PROGRAMMING_LAB_PYTHON/CO5-Q2.py ====
['ANANYA\n', 'AMMU\n', 'ANU\n', 'RESHMA\n']
>>> 
```

PROGRAM NO : 40

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

ALGORITHM:

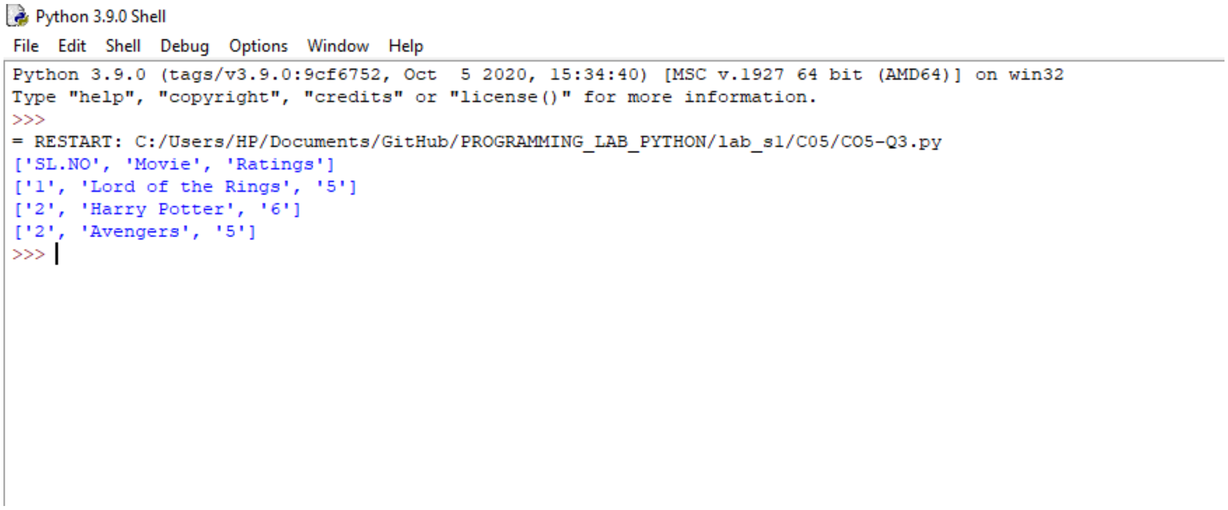
```
Step1: Start
Step2: First import csv
Step3: Open a file named movie1.csv
Step4: Write datas in different rows using writer.writerow()
Step5: After writing data ,again open the file movie1.csv
Step6: Then read each row from movie1.csv and display the list of strings
Step7: Stop
```

PROGRAM:

```
import csv
with open('movie1.csv', 'w', newline='') as file:
    writer = csv.writer(file)
    writer.writerow(["SL.NO", "Movie", "Ratings"])
    writer.writerow([1, "Lord of the Rings", 5])
    writer.writerow([2, "Harry Potter", 6])
    writer.writerow([2, "Avengers", 5])
with open('movie1.csv', 'r') as file:
    reader = csv.reader(file)
    for row in reader:
        print(row)
```

RESULT: The above program is successfully executed and obtained the output

OUTPUT



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:/Users/HP/Documents/GitHub/PROGRAMMING_LAB_PYTHON/lab_s1/C05/CO5-Q3.py
['SL.NO', 'Movie', 'Ratings']
['1', 'Lord of the Rings', '5']
['2', 'Harry Potter', '6']
['2', 'Avengers', '5']
>>> 
```

PROGRAM NO : 41

AIM: Write a Python program to read specific columns of a given CSV file and print the content of the columns.

ALGORITHM:

Step1: Start

Step2: import csv and opened a file movie2.csv in write mode

Step3: Write 5 rows of data into that csv file

Step4: Then again opened that file in read mode and

Step5: Read the content in column ‘ Movie’ and displayed that data

Step6: Stop

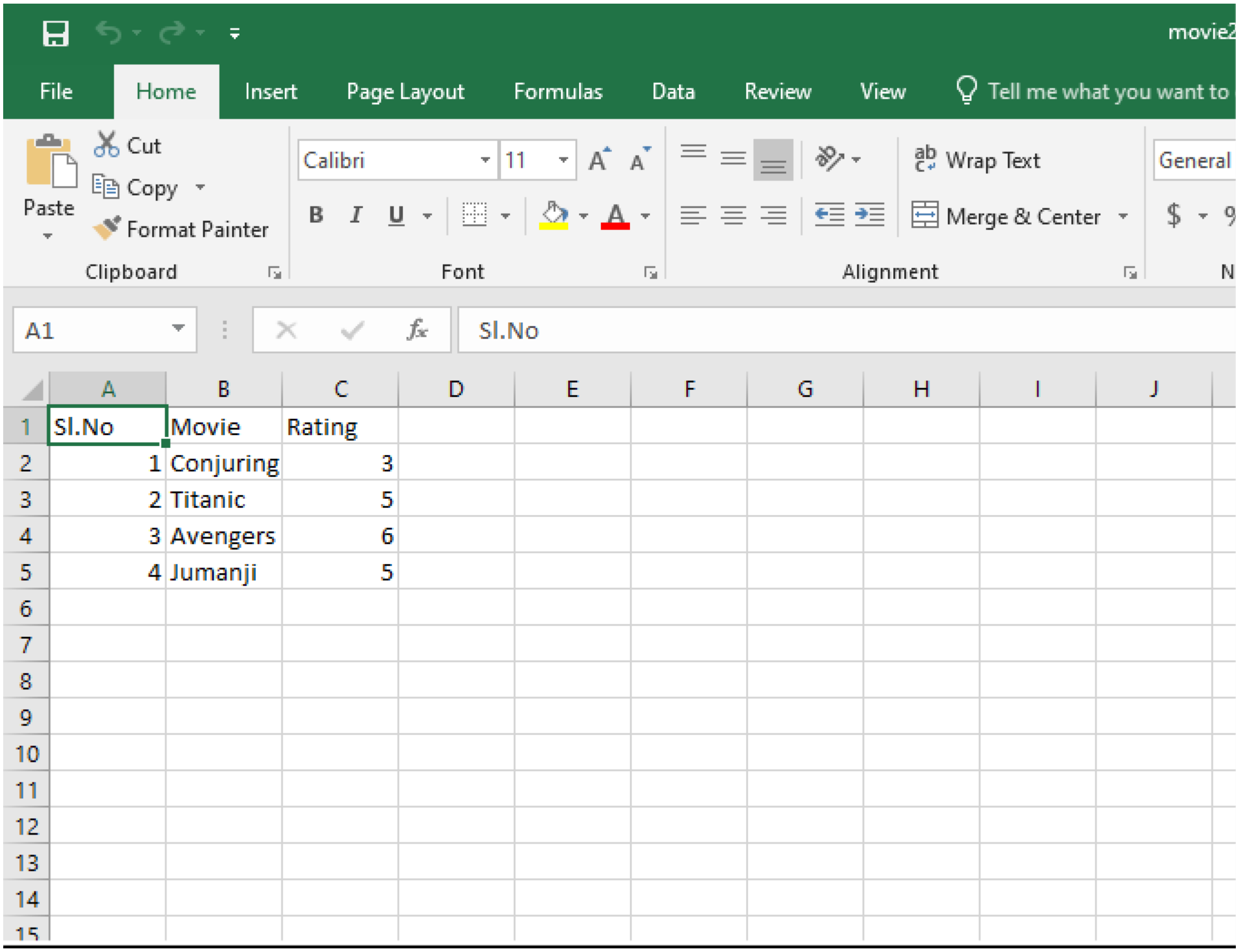
PROGRAM:

```
import csv
with open("movie2.csv",'w',newline='') as file:
  write=csv.writer(file)
  write.writerow(["Sl.No","Movie","Rating"])
  write.writerow(["1","Conjuring 2","3"])
  write.writerow(["2","Titanic","5"])
  write.writerow(["3","Avengers","6"])
  write.writerow(["4","Jumanji","5"])
with open("movie2.csv" ,'r') as file:
    data=csv.reader(file)
    print("Contents in Column 'Movie': ")
    for r in data:
        print(r[1])
```

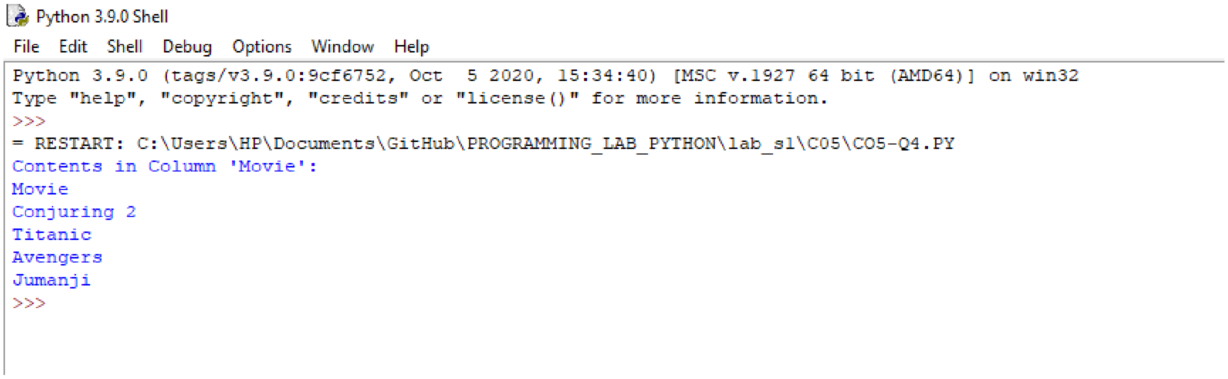
RESULT: The above program is successfully executed and obtained the output

OUTPUT:

movie2.csv



	A	B	C	D	E	F	G	H	I	J
1	Sl.No	Movie	Rating							
2	1	Conjuring	3							
3	2	Titanic	5							
4	3	Avengers	6							
5	4	Jumanji	5							



```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct  5 2020, 15:34:40) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\HP\Documents\GitHub\PROGRAMMING_LAB_PYTHON\lab_s1\C05\C05-Q4.PY
Contents in Column 'Movie':
Movie
Conjuring 2
Titanic
Avengers
Jumanji
>>>
```

PROGRAM NO : 42

AIM: Write a Python program to write a Python dictionary to a csv file. After writing the CSV file read the CSV file and display the content.

ALGORITHM:

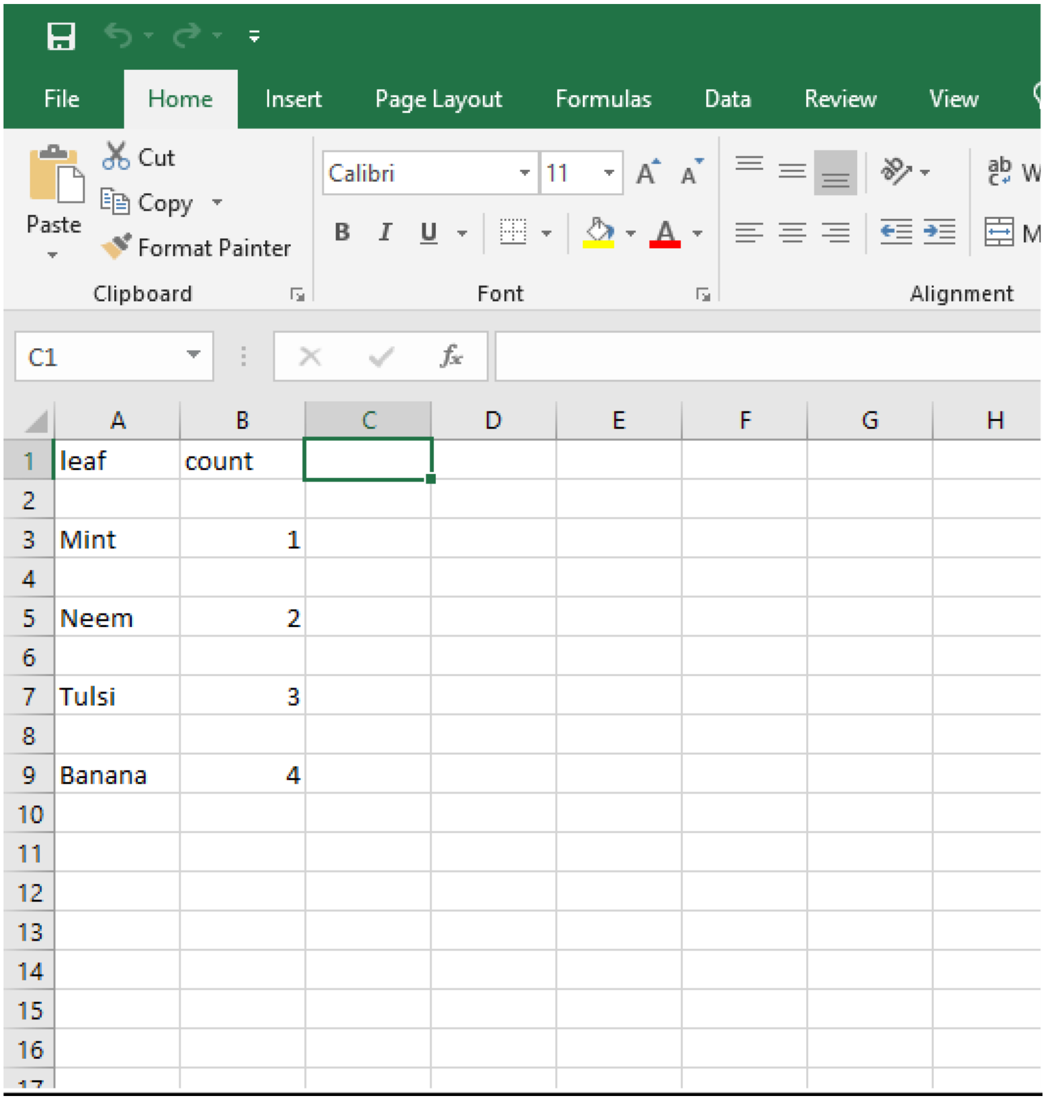
```
Step1: Start
Step2: Open a csv file named leaves.csv in write mode
Step3: Write data to 4 rows of csv file to dictionary
Step4: Again open leaves .csv and read the contents in the file and displayed each row of data
Step5: Stop
```

PROGRAM:

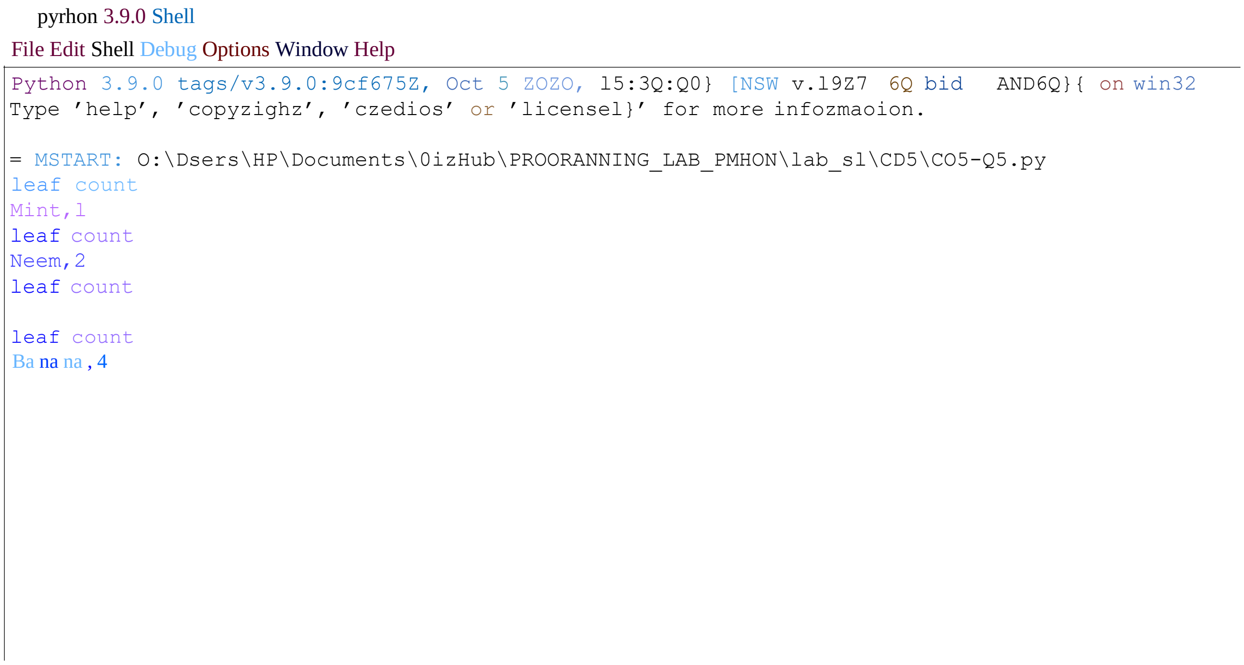
```
import csv
f=open("leaves.csv","w")
writer=csv.DictWriter(f,fieldnames=["leaf","count"])
writer.writeheader()
writer.writerow({"leaf":"Mint","count":"1"})
writer.writerow({"leaf":"Neem","count":"2"})
writer.writerow({"leaf":"Tulsi","count":"3"})
writer.writerow({"leaf":"Banana","count":"4"})
f.close()
c=0
f=open("leaves.csv")
reader=csv.DictReader(f)
for row in reader:
    if c==0:
        print(f'{" ".join(row)}')
    print(f'{row["leaf"]},{row["count"]}')
f.close()
```

RESULT: The above program is successfully executed and obtained the o

OUTPUT



	A	B
1	leaf	count
2		
3	Mint	1
4		
5	Neem	2
6		
7	Tulsi	3
8		
9	Banana	4



```
Python 3.9.0 tags/v3.9.0:9cf675Z, Oct 5 ZOZO, l5:3Q:Q0} [NSW v.l9Z7  6Q bid   AND6Q}{ on win32
Type ’help’, ’copyzighz’, ’czedios’ or ’licensel}’ for more infozmaoion.

= MSTART: O:\Dsers\HP\Documents\0izHub\PROORANNING_LAB_PMHON\lab_sl\CD5\CO5-Q5.py
leaf count
Mint,l
leaf count
Neem,2
leaf count

leaf count
Ba na na , 4
```