COURSE OUTCOME 1 (CO1)

PROGRAM NO: 1 **DATE: 03-11-2021**

AIM : Familiarizing Text Editor, IDE, Code Analysis Tools etc. // Use any IDE

It is a Graphical User Interface (GUI) where programmers write their code and produce the final products. An IDE basically unifies all essential tools required for software development and testing, which in turn helps the programming maximize his output.

➢ Features of IDE:-

1. Code Editor
2. Syntax Highlighting
3. Auto completion code
4. Debugger
5. Compiler
6. Language Support

IDLE is Python's Integrated Development and Learning Environment.

IDLE has the following features:

- coded in 100% pure Python, using the tkinter GUI toolkit.
- cross-platform: works mostly the same on Windows, Unix, and macOS.
- Python shell window (interactive interpreter) with colorizing of code input, output, and error messages.
- multi-window text editor with multiple undo, Python colorizing, smart indent, call tips, auto completion, and other features.
- search within any window, replace within editor windows, and search through multiple files (grep).
- debugger with persistent breakpoints, stepping, and viewing of global and local namespaces.
- configuration, browsers, and other dialogs.

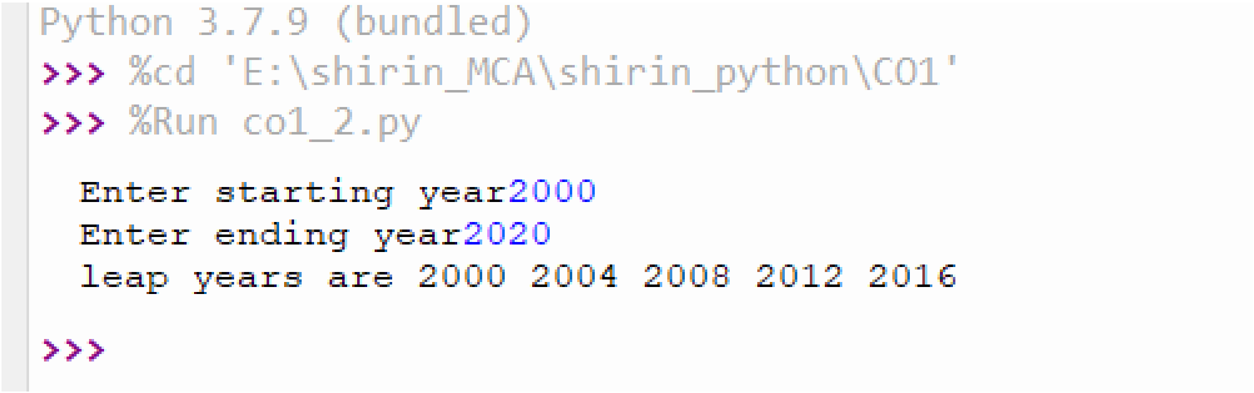
PROGRAM NO: 2 **DATE: 08-11-2021**

AIM : Write a program to Find Leap Year.

PROGRAM

```
s=int(input("Enter starting year"))

e=int(input("Enter ending year"))

if(s<e):

    print("leap years are",end=" ")

  for i in range(s,e):

        if(i%4==0 and i%100!=0 or i%400==0 and i%100==0):

            print(i,end=" ")
```

OUTPUT



```
Python 3.7.9 (bundled)
>>> %cd 'E:\shirin_MCA\shirin_python\CO1'
>>> %Run co1_2.py

  Enter starting year2000
  Enter ending year2020
  leap years are 2000 2004 2008 2012 2016

>>>
```

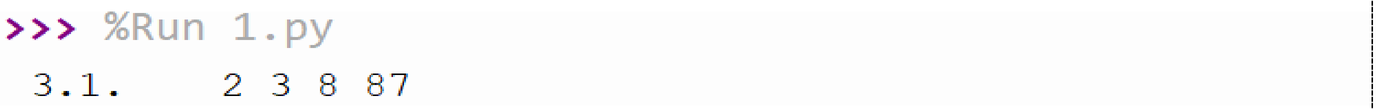
PROGRAM NO: 3 **DATE: 10-11-2021**

AIM : 3.1. Generate positive list of numbers from a given list of integers

PROGRAM

```
print("3.1.   ",end=" ")

for i in [-1,2,3,-87,8,87,-9]:

   if(i>0):

      print(I, end=" "
```

OUTPUT



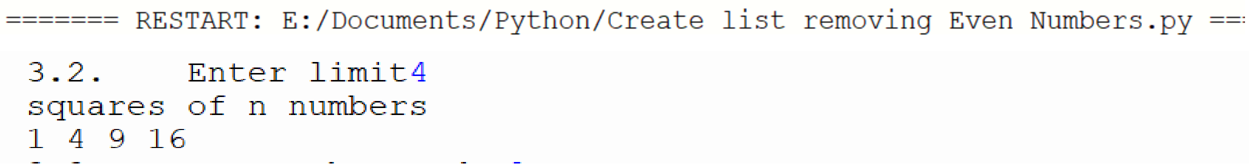
```
>>> %Run 1.py
 3.1.    2 3 8 87
```

AIM : 3.2. Write a program to find the Square of N number

PROGRAM

```
n=int(input("3.2.    Enter limit"))

i=1

print("squares of n numbers")

while(i<=n):

   print(i*i,end=" ")

   i=i+1
```

OUTPUT



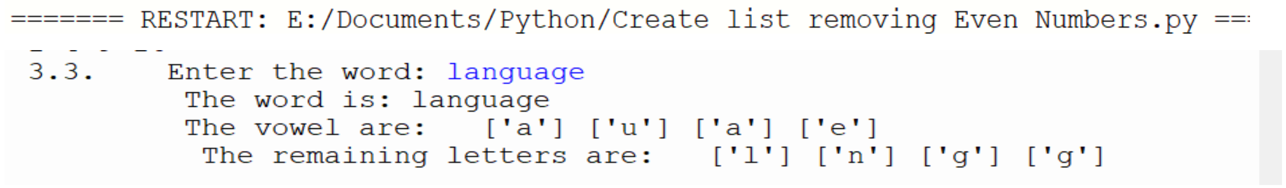
```
======= RESTART: E:/Documents/Python/Create list removing Even Numbers.py ==
 3.2.    Enter limit4
 squares of n numbers
 1 4 9 16
```

AIM : 3.3. Form a list of vowels selected from a given word

PROGRAM

```
n=str(input("3.3.    Enter the word: "))
print("         The word is: "+n)
print("         The vowel are:  " ,end=" ")
for i in n:
    if i in 'aeiouAEIOU':
        print([i],end=" ")
print(" \n          The remaining letters are:  ",end=" ")
for j in n:
 if j not in 'aeiouAEIOU':
        print([j],end=" ")
print()
```

OUTPUT



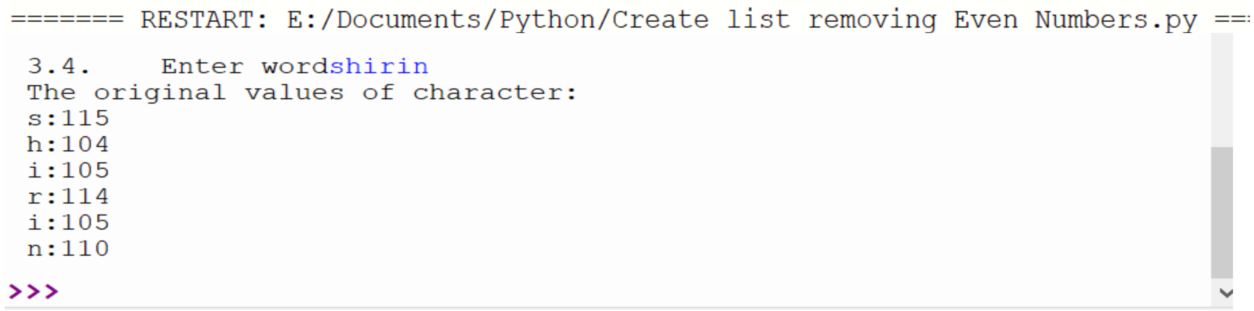
```
======= RESTART: E:/Documents/Python/Create list removing Even Numbers.py ==
 3.3.    Enter the word: language
          The word is: language
          The vowel are:    ['a'] ['u'] ['a'] ['e']
           The remaining letters are:    ['l'] ['n'] ['g'] ['g']
```

AIM : 3.4. List ordinal value of each element of a word (Hint: use ord() to get ordinal values)

PROGRAM

```
word=input("3.4.    Enter word")
print("The original values of character:")
for i in word:
    print(i,end=":")
    print(ord(i))
```

OUTPUT



```
======= RESTART: E:/Documents/Python/Create list removing Even Numbers.py ==

 3.4.    Enter wordshirin
 The original values of character:
 s:115
 h:104
 i:105
 r:114
 i:105
 n:110
>>>
```

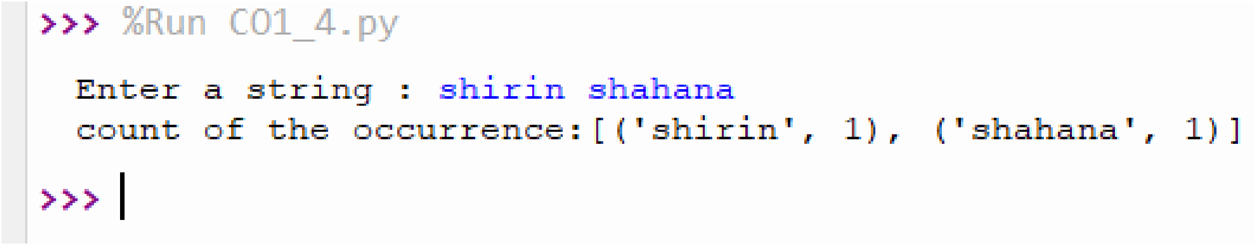
PROGRAM NO: 4 **DATE: 15-11-2021**

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

PROGRAM

```
str1 = input("Enter a string : ")

wordlist = str1.split()

count= []

for w in wordlist: count.append(wordlist.count(w))

print("count of the occurrence:" + str(list(zip(wordlist, count))))
```

OUTPUT



```
>>> %Run CO1_4.py

  Enter a string : shirin shahana
  count of the occurrence:[('shirin', 1), ('shahana', 1)]

>>> 
```

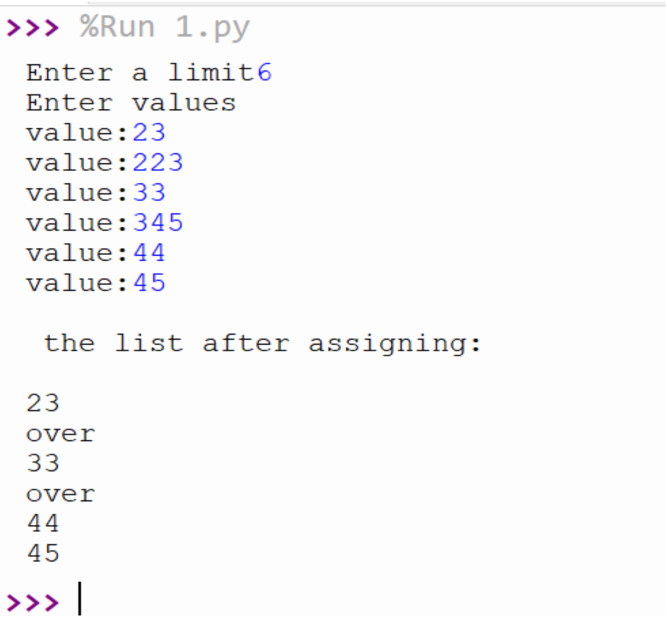
PROGRAM NO: 5 **DATE: 15-11-2021**

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

PROGRAM

```
n=[]
s=int(input("Enter a limit"))
print("Enter values")
i=0
while(i<s):
   num=input("value:")
   n.append(int(num))
   i=i+1
print("\n the list after assigning:\n")
i=0
while(i<len(n)):
   if(n[i]>100):
      print("over")
   else:
      print(n[i])
   i=i+1
```

OUTPUT



```
>>> %Run 1.py
  Enter a limit6
  Enter values
  value:23
  value:223
  value:33
  value:345
  value:44
  value:45

   the list after assigning:

  23
  over
  33
  over
  44
  45

>>> |
```

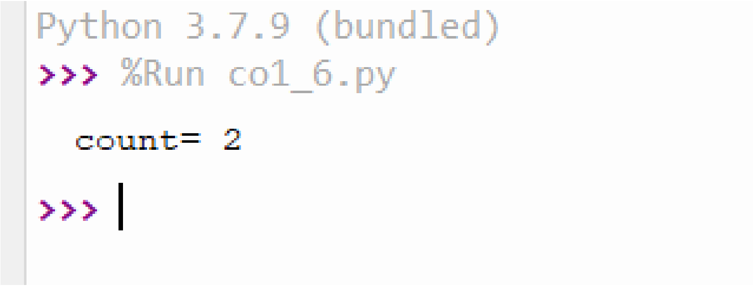
PROGRAM NO: 6 **DATE: 17-11-2021**

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

PROGRAM

```
list1=['a','b','s','a']

occ=list1.count('a')

print("count=",occ)
```

OUTPUT



```
Python 3.7.9 (bundled)
>>> %Run co1_6.py

  count= 2

>>> 
```

PROGRAM NO: 7 **DATE: 17-11-2021**

AIM : Enter 2 lists of integers. Check

(a) Whether list are of same length.

(b) whether list sums to same value.

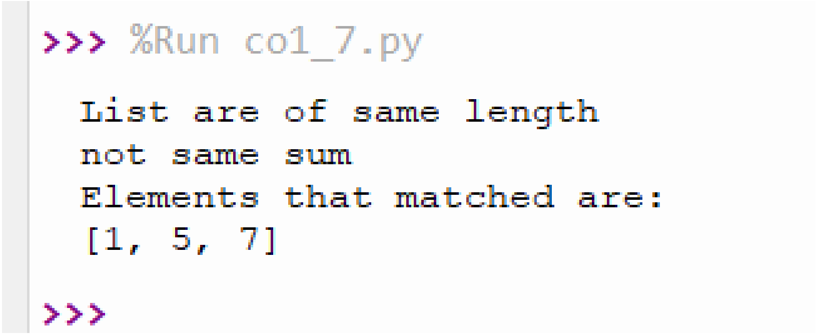
(c) whether any value occur in both.

PROGRAM

```
lst=[1,3,5,7,9,11,34]
lst1=[5,13,45,7,20,65,1]
s=int(0)
c=int(0)
if(len(lst)==len(lst1)):
    print("List are of same length")
else:
    print("list have different length")
for i in range(0,len(lst) and len(lst1)):
    s=s+lst[i]
    c=c+lst1[i]
if(s==c):
  print("equal sum")
else:
  print("not same sum")

print("Elements that matched are:")
l=[]
for i in range(0,len(lst)):
  for j in range(0,len(lst1)):
```

```
 if lst[i]==lst1[j]:
        l.append(lst[i] and lst1[j])
    else:
      continue
print(l)
```

OUTPUT



```
>>> %Run co1_7.py
  List are of same length
  not same sum
  Elements that matched are:
  [1, 5, 7]

>>>
```

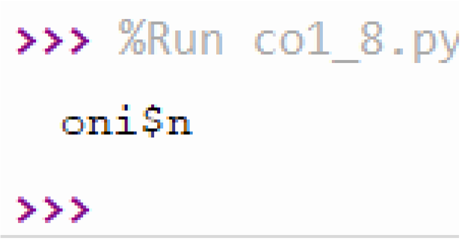
PROGRAM NO: 8 **DATE: 22-11-2021**

AIM : Get a string from an input string where all occurrences of first character replaced with '$', except first character. [eg: onion -> oni$n]

PROGRAM

```
str='onion'

char=str[0]

str=str.replace(char,'$')

print(char+str[1:])
```

OUTPUT



```
>>> %Run co1_8.py
  oni$n
>>>
```

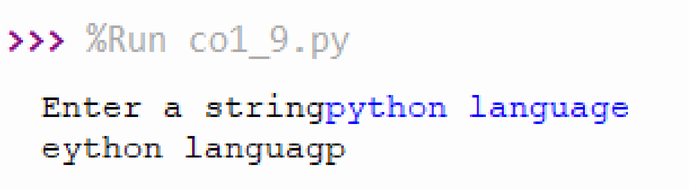
PROGRAM NO: 9 **DATE: 22-11-2021**

AIM : Create a string from given string where first and last characters exchanged.
[eg: python -> nythop]

PROGRAM

```
str=input("Enter a string")

print(str[-1]+str[1:-1]+str[0])
```

OUTPUT



```
>>> %Run co1_9.py

  Enter a stringpython language
  eython languagp
```

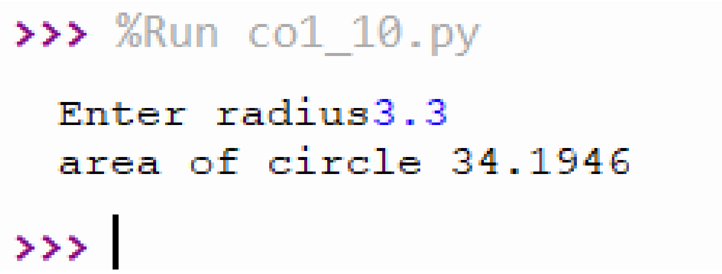
PROGRAM NO: 10 **DATE: 24-11-2021**

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

PROGRAM

```
p=3.14

rad=float(input("Enter radius"))

area=p*rad*rad

print("area of circle",area)
```

OUTPUT



```
>>> %Run co1_10.py

  Enter radius3.3
  area of circle 34.1946

>>> 
```

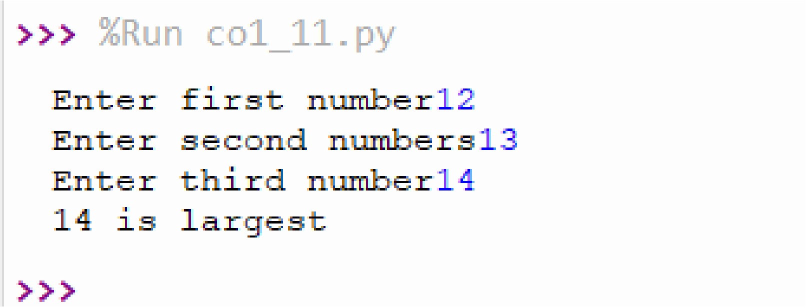
PROGRAM NO: 11 **DATE: 24-11-2021**

AIM : Write a program to find biggest of 3 numbers entered.

PROGRAM

```
a=int(input("Enter first number"))

b=int(input("Enter second numbers"))

c=int(input("Enter third number"))

if(a>b and a>c):

   print(a,"is largest")

elif(b>c):

   print(b,"is largest")

else:

   print(c,"is largest")
```

OUTPUT



```
>>> %Run co1_11.py

  Enter first number12
  Enter second numbers13
  Enter third number14
  14 is largest

>>>
```

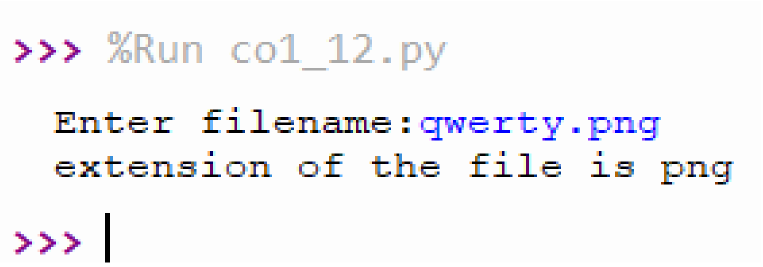
PROGRAM NO: 12 **DATE: 24-11-2021**

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

PROGRAM

```
file=input("Enter filename:")

f=file.split(".")

print("extension of the file is "+f[-1])
```

OUTPUT



```
>>> %Run co1_12.py

  Enter filename:qwerty.png
  extension of the file is png

>>> 
```

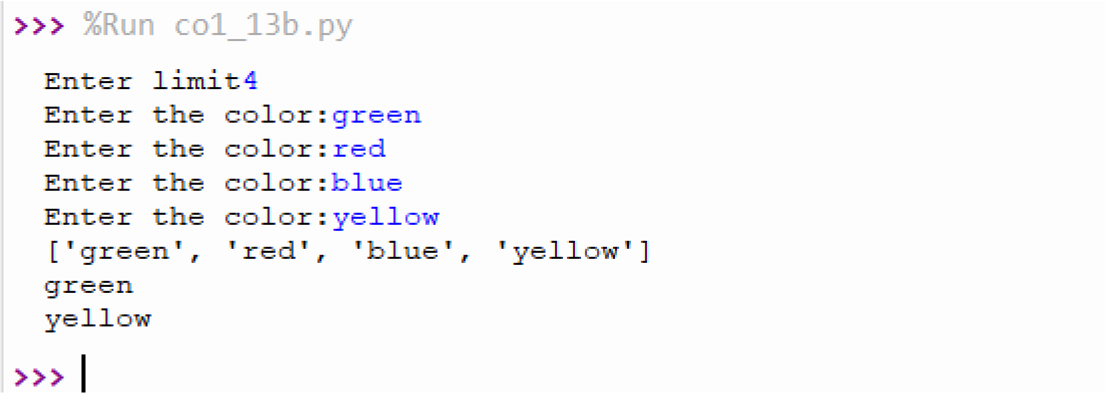
PROGRAM NO: 13 **DATE: 29-11-2021**

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

PROGRAM

```
a=[]

n=int(input("Enter limit"))

for i in range(0,n):

   b=input("Enter the color:")

   a.append(b)

print(a)

print(a[0])

print(a[n-1])
```

OUTPUT



```
>>> %Run co1_13b.py

  Enter limit4
  Enter the color:green
  Enter the color:red
  Enter the color:blue
  Enter the color:yellow
  ['green', 'red', 'blue', 'yellow']
  green
  yellow

>>> 
```

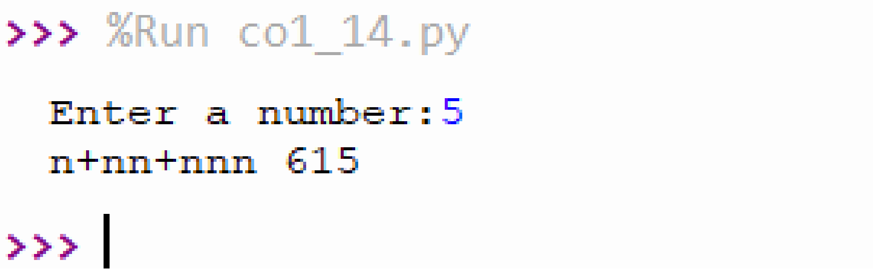
PROGRAM NO: 14 **DATE: 29-11-2021**

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

PROGRAM

```
n=int(input("Enter a number:"))

x=int("%s"%n)

y=int("%s%s"%(n,n))

z=int("%s%s%s"%(n,n,n))

print("n+nn+nnn",x+y+z)
```

OUTPUT



```
>>> %Run co1_14.py

  Enter a number:5
  n+nn+nnn 615

>>> 
```

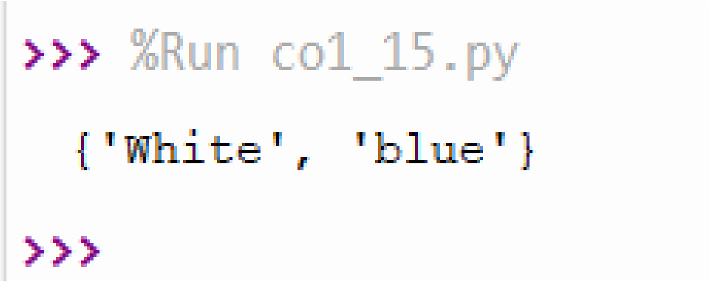
PROGRAM NO: 15 **DATE: 29-11-2021**

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

PROGRAM

```
color_list_1=set(["White","pink","blue"])

color_list_2=set(["red","green","pink"])

print(color_list_1.difference(color_list_2))
```

OUTPUT



```
>>> %Run co1_15.py

  {'White', 'blue'}

>>>
```

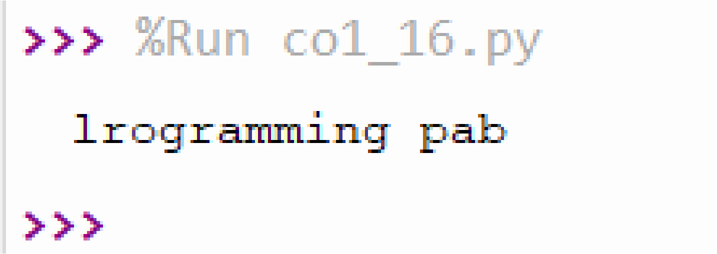
PROGRAM NO: 16 **DATE: 29-11-2021**

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

PROGRAM

```
a="programming"

b="lab"

#r1=a[0]

#r2=b[0]

print(b[0]+a[1:]+" "+a[0]+b[1:])
```

OUTPUT



```
>>> %Run co1_16.py
  lrogramming pab
>>>
```

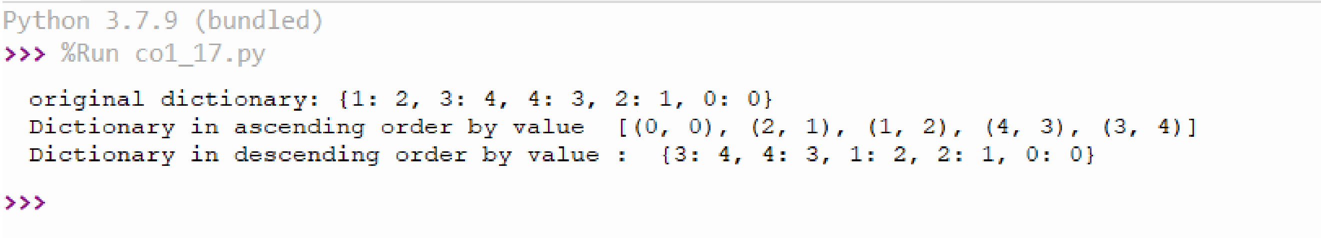
PROGRAM NO: 17 **DATE: 01-12-2021**

AIM : Sort dictionary in ascending and descending order.

PROGRAM

```
import operator

d={1:2,3:4,4:3,2:1,0:0}

print("original dictionary:",d)

sorted_d = sorted(d.items(), key=operator.itemgetter(1))

print('Dictionary in ascending order by value ',sorted_d)

sorted_d = dict( sorted(d.items(),key=operator.itemgetter(1),reverse=True))

print('Dictionary in descending order by value : ',sorted_d)
```

OUTPUT



```
Python 3.7.9 (bundled)
>>> %Run co1_17.py

  original dictionary: {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}
  Dictionary in ascending order by value  [(0, 0), (2, 1), (1, 2), (4, 3), (3, 4)]
  Dictionary in descending order by value :  {3: 4, 4: 3, 1: 2, 2: 1, 0: 0}

>>>
```

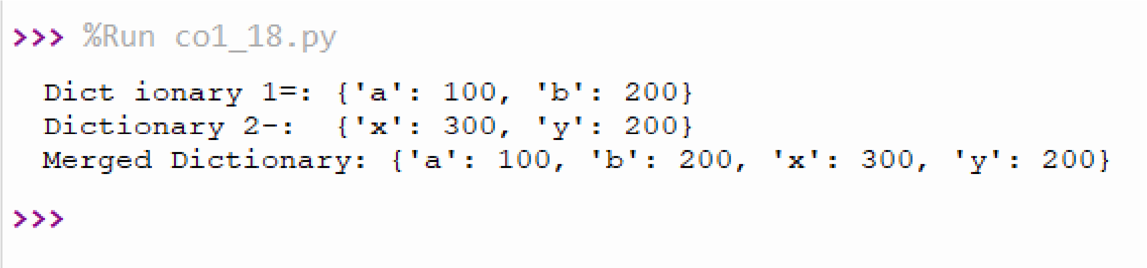
PROGRAM NO: 18 **DATE: 01-12-2021**

AIM : Write a program to merge two dictionaries.

PROGRAM

```
d1 ={ 'a': 100, 'b': 200}

d2 ={'x' : 300, 'y': 200}

print ("Dict ionary 1=:", d1)

print ("Dictionary 2-: ", d2)

d1.update(d2)

print("Merged Dictionary:",d1)
```

OUTPUT



```
>>> %Run co1_18.py

  Dict ionary 1=: {'a': 100, 'b': 200}
  Dictionary 2-:  {'x': 300, 'y': 200}
  Merged Dictionary: {'a': 100, 'b': 200, 'x': 300, 'y': 200}

>>>
```

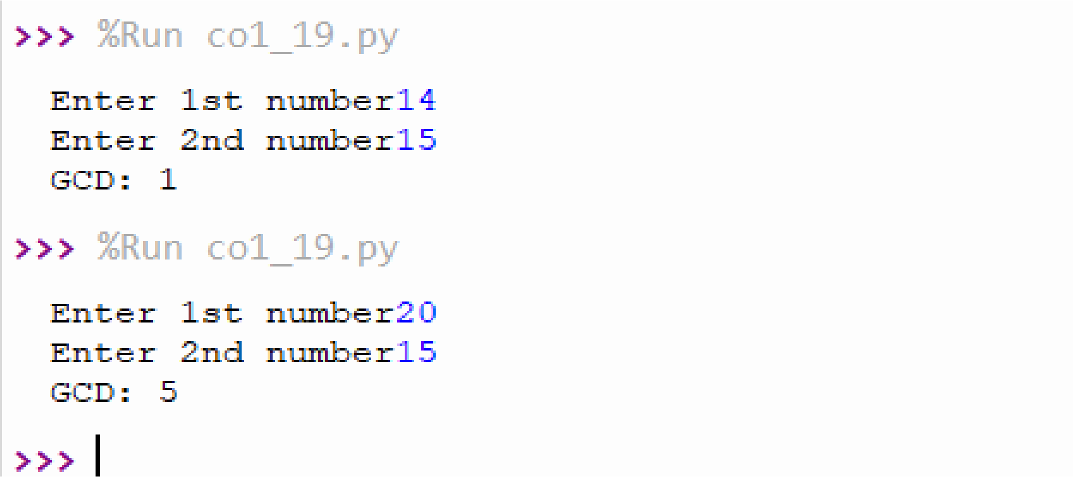
PROGRAM NO: 19 **DATE: 01-12-2021**

AIM : Write a program to find GCD of 2 numbers.

PROGRAM

```
a=int(input("Enter 1st number"))

b=int(input("Enter 2nd number"))

i=1

while(i<=a and i<=b):

    if(a%i==0 and b%i==0):

        gcd=i

    i=i+1

print("GCD:",gcd)
```

OUTPUT



```
>>> %Run co1_19.py

  Enter 1st number14
  Enter 2nd number15
  GCD: 1

>>> %Run co1_19.py

  Enter 1st number20
  Enter 2nd number15
  GCD: 5

>>> 
```

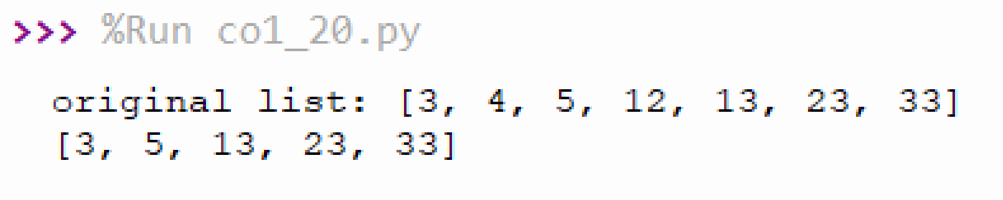
PROGRAM NO: 20 **DATE: 01-12-2021**

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

PROGRAM

```
num=[3,4,5,12,13,23,33]

n=[]

print("original list:",num)

for i in num:

    if(i%2!=0):

        n.append(i)

print(n)
```

OUTPUT



```
>>> %Run co1_20.py

  original list: [3, 4, 5, 12, 13, 23, 33]
  [3, 5, 13, 23, 33]
```

COURSE OUTCOME 2 (CO2)

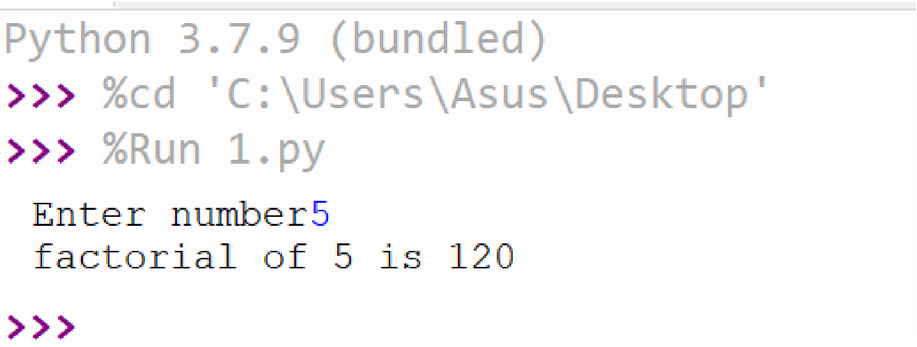
PROGRAM NO: 1 **DATE: 06-12-2021**

AIM : Write a program to find the factorial of a number.

PROGRAM

```
n=int(input("Enter number"))

f=1

for i in range(1,n+1):

   f=f*i

print("factorial of",n,"is",f)
```

OUTPUT



```
Python 3.7.9 (bundled)
>>> %cd 'C:\Users\Asus\Desktop'
>>> %Run 1.py

 Enter number5
 factorial of 5 is 120

>>>
```

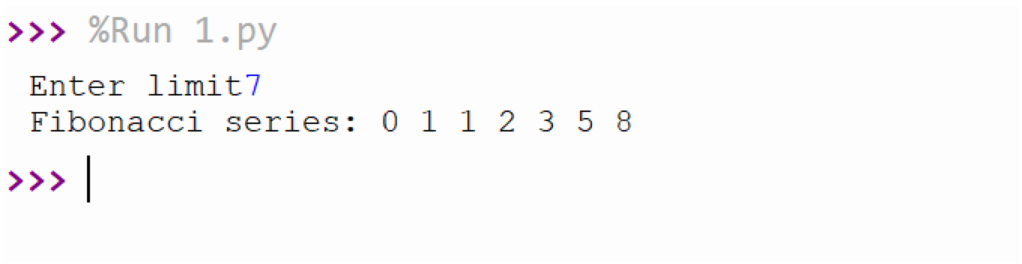
PROGRAM NO: 2 **DATE: 06-12-2021**

AIM : Write a program to generate Fibonacci series of N terms.

PROGRAM

```
n=int(input("Enter limit"))

a=0

b=1

sum=0

print("Fibonacci series:",end=" ")

i=1

while(i<=n):

    print(sum,end=" ")

    a=b

    b=sum

    sum=a+b

    i=i+1
```

OUTPUT



```
>>> %Run 1.py
 Enter limit7
 Fibonacci series: 0 1 1 2 3 5 8
>>> 
```

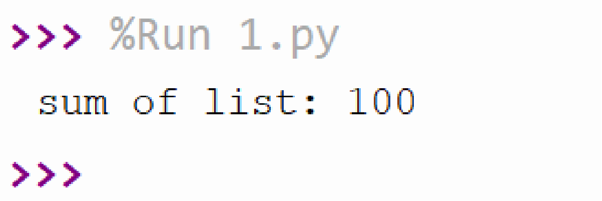
PROGRAM NO: 3 **DATE: 06-12-2021**

AIM : Write a program to find the sum of all items in a list.

PROGRAM

```
list1=[10,15,20,25,30]

total=sum(list1)

print("sum of list:",total)
```

OUTPUT



```
>>> %Run 1.py
 sum of list: 100
>>>
```

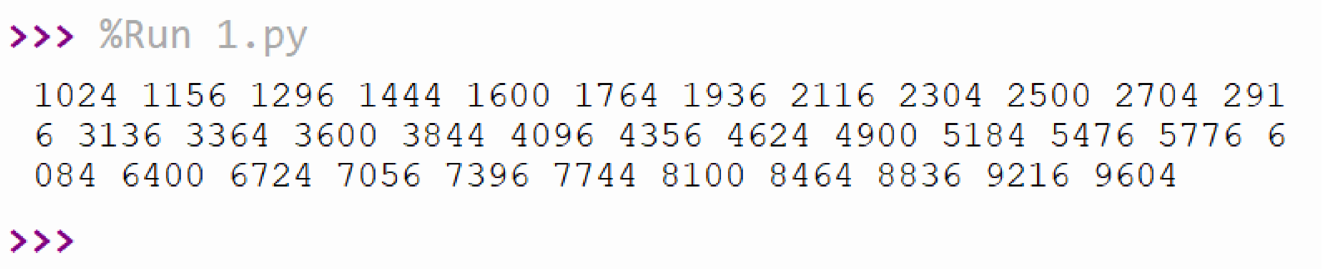
PROGRAM NO: 4 **DATE: 06-12-2021**

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

PROGRAM

```
from math import sqrt as s

for i in range(1000,10000):

    if s(i)==int(s(i)) and i%2==0:

        print(i,end=" ")
```

OUTPUT



```
>>> %Run 1.py
  1024 1156 1296 1444 1600 1764 1936 2116 2304 2500 2704 291
  6 3136 3364 3600 3844 4096 4356 4624 4900 5184 5476 5776 6
  084 6400 6724 7056 7396 7744 8100 8464 8836 9216 9604
>>>
```

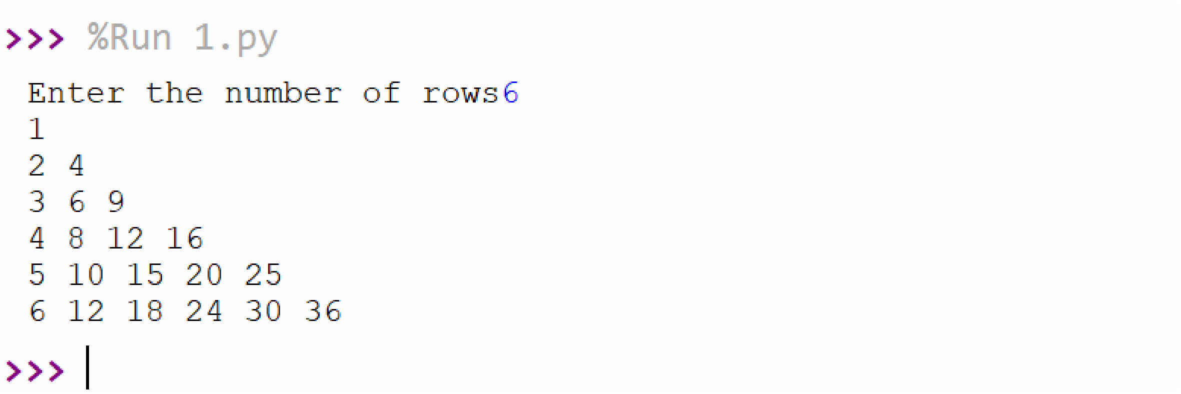
PROGRAM NO: 5 **DATE: 06-12-2021**

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

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()
```

OUTPUT



```
>>> %Run 1.py
 Enter the number of rows6
 1
 2 4
 3 6 9
 4 8 12 16
 5 10 15 20 25
 6 12 18 24 30 36
>>> 
```

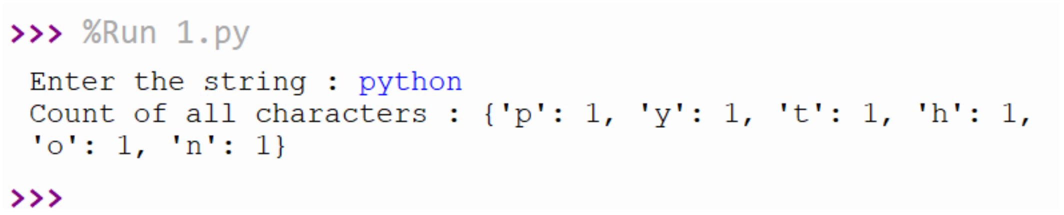
PROGRAM NO: 6 **DATE: 06-12-2021**

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

PROGRAM

```
test_str=str(input("Enter the string : "))

freq = {}

for i in test_str:

    if i in freq:

        freq[i] += 1

    else:

        freq[i] = 1

print ("Count of all characters : "+ str(freq))
```

OUTPUT



```
>>> %Run 1.py
 Enter the string : python
 Count of all characters : {'p': 1, 'y': 1, 't': 1, 'h': 1,
 'o': 1, 'n': 1}
>>>
```

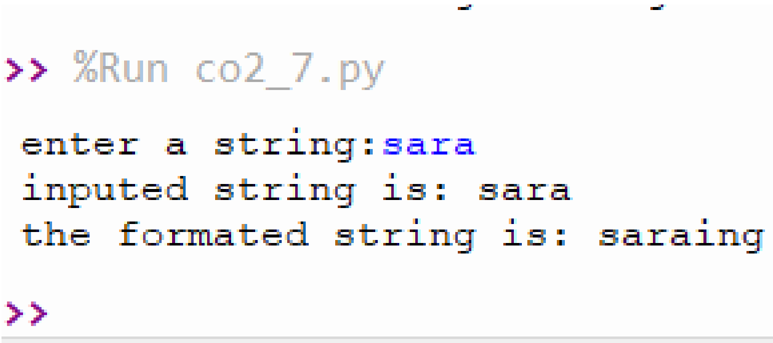
PROGRAM NO: 7 **DATE: 08-12-2021**

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

PROGRAM

```
str=input("enter a string:")

print("inputed string is:",str)

if(str.endswith("ing")):

    str=str+'ly'

else:

    str=str+'ing'

print("the formated string is:",str)
```

OUTPUT



```
>> %Run co2_7.py

 enter a string:sara
 inputed string is: sara
 the formated string is: saraing

>>
```

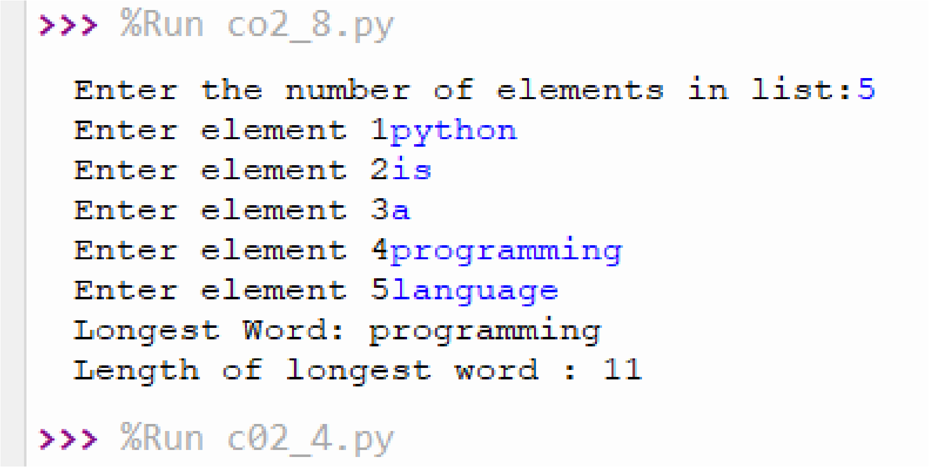
PROGRAM NO: 8 **DATE: 08-12-2021**

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

PROGRAM

```
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
   element=input("Enter element "+ str(x+1) )
   a.append(element)
   max1=len(a[0])
   temp=a[0]
for i in a:
   if(len(i)>max1):
      max1=len(i)
      temp=i
print("Longest Word:",temp)
print("Length of longest word :",max1)
```

OUTPUT



```
>>> %Run co2_8.py
  Enter the number of elements in list:5
  Enter element 1python
  Enter element 2is
  Enter element 3a
  Enter element 4programming
  Enter element 5language
  Longest Word: programming
  Length of longest word : 11
>>> %Run c02_4.py
```

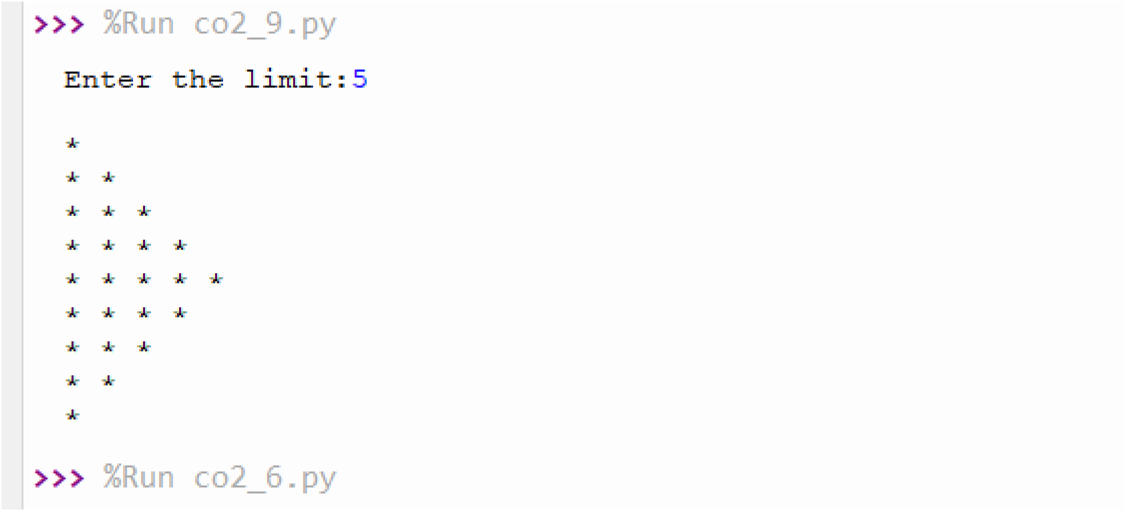
PROGRAM NO: 9 **DATE: 08-12-2021**

AIM : Construct following pattern using nested loop

PROGRAM

```
n= int(input("Enter the limit:"))
for i in range(n):
    for j in range(i):
        print ('* ', end="")
    print('')
for i in range(n,0,-1):
    for j in range(i):
        print('* ', end="")
    print('')
```

OUTPUT



```
>>> %Run co2_9.py
  Enter the limit:5

  *
  * *
  * * *
  * * * *
  * * * * *
  * * * *
  * * *
  * *
  *

>>> %Run co2_6.py
```

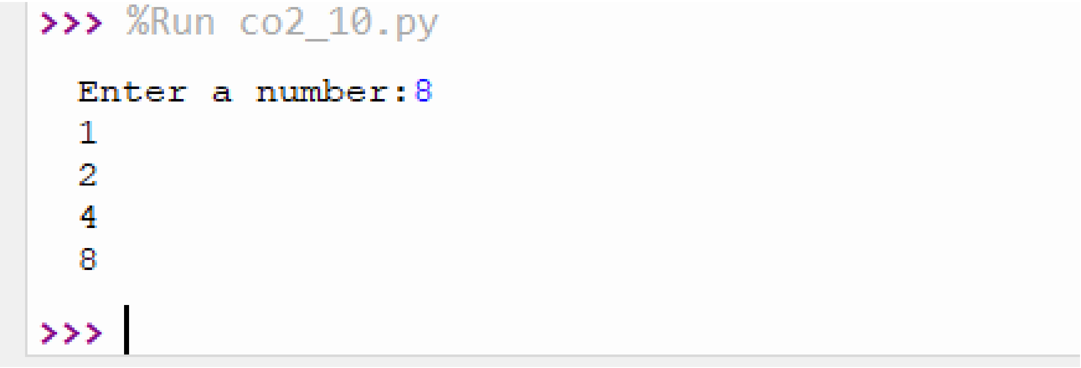
PROGRAM NO: 10 **DATE: 08-12-2021**

AIM : Generate all factors of a number. def print_factors(x):

PROGRAM

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

OUTPUT



```
>>> %Run co2_10.py

  Enter a number:8
  1
  2
  4
  8

>>> 
```

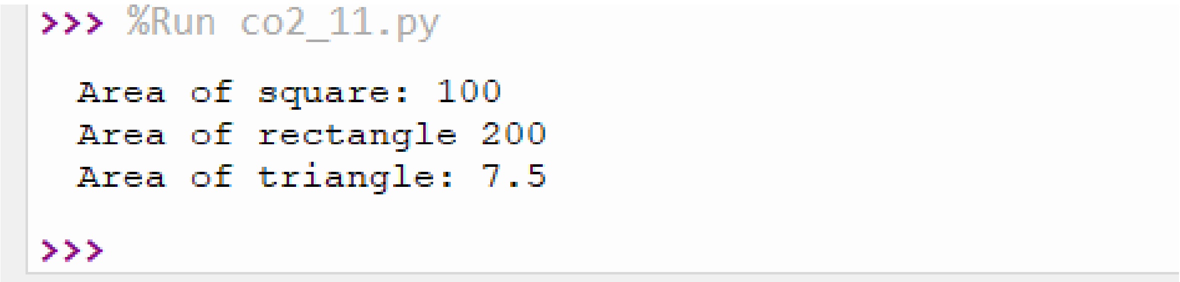
PROGRAM NO: 11 **DATE: 08-12-2021**

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

PROGRAM

```
s_area=lambda a:a*a

r_area=lambda a,b:a*b

t_area=lambda b,h:1/2*b*h

print("Area of square:",s_area(10))

print("Area of rectangle",r_area(10,20))

print("Area of triangle:",t_area(5,3))
```

OUTPUT



```
>>> %Run co2_11.py

  Area of square: 100
  Area of rectangle 200
  Area of triangle: 7.5

>>>
```

COURSE OUTCOME 3 (CO3)

PROGRAM NO: 1 **DATE: 13-12-2021**

AIM : Work with built-in packages.

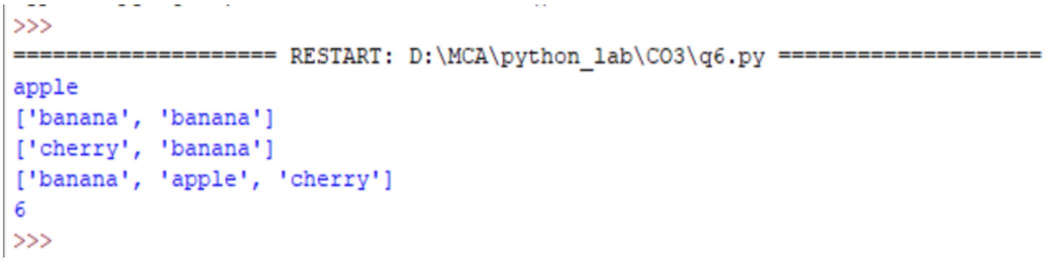
A) **random module.**
B) **time module**
C) **calendar module**
D) **math module**
E) **Statistics module**

A) random module

PROGRAM

```
import random

mylist = ["apple", "banana", "cherry"]

print(random.choice(mylist))        #Returns a random element from the given sequence

print(random.choices(mylist, k=2))

print(random.sample(mylist, k=2))   #Return a list that contains any 2 of the items from a list:

random.shuffle(mylist)

print(mylist)                   #Takes a sequence and returns the sequence in a random order

print(random.randrange(3, 9))       #Return a number between 3 and 9:
```

OUTPUT



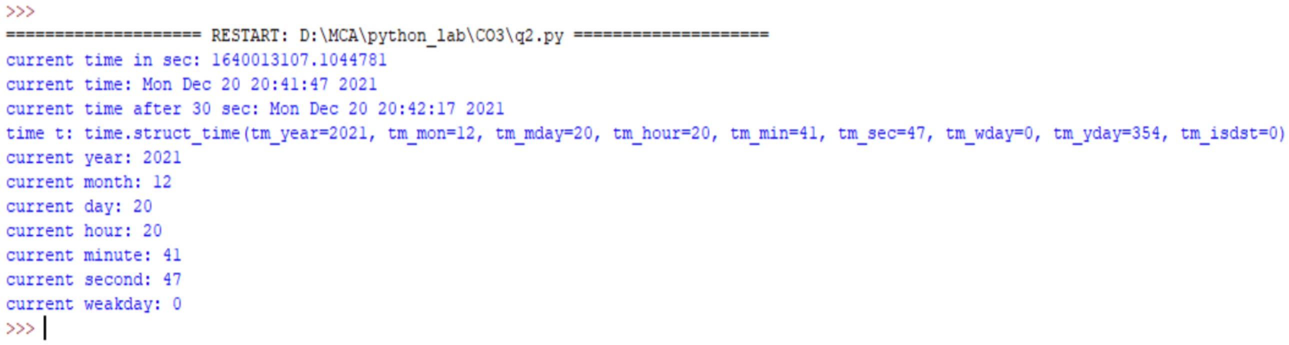
```
>>>
==================== RESTART: D:\MCA\python_lab\CO3\q6.py ====================
apple
['banana', 'banana']
['cherry', 'banana']
['banana', 'apple', 'cherry']
6
>>>
```

B) time module

PROGRAM

```
import time

print("current time in sec:",time.time())

print("current time:",time.ctime())

print("current time after 30 sec:",time.ctime(time.time()+30))

t=time.localtime()

print("time t:",t)

print("current year:",t.tm_year)

print("current month:",t.tm_mon)

print("current day:",t.tm_mday)

print("current hour:",t.tm_hour)

print("current minute:",t.tm_min)

print("current second:",t.tm_sec)

print("current weakday:",t.tm_wday)
```

OUTPUT



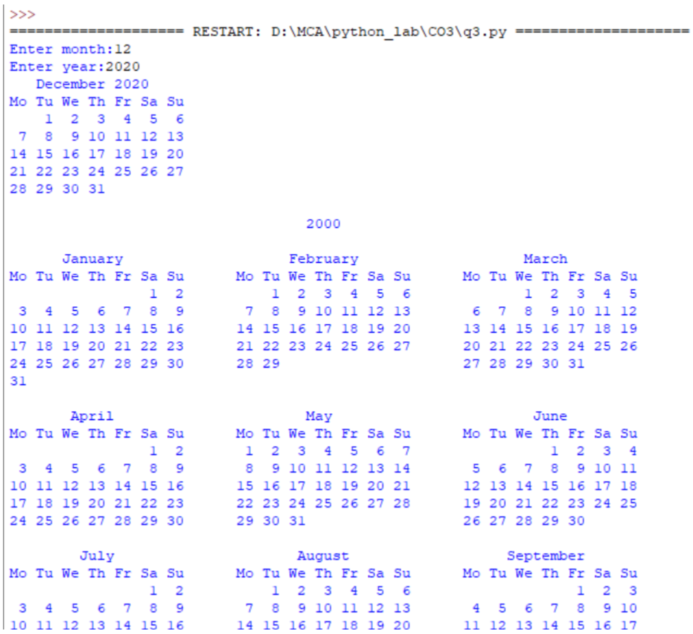
```
>>>
==================== RESTART: D:\MCA\python_lab\CO3\q2.py ====================
current time in sec: 1640013107.1044781
current time: Mon Dec 20 20:41:47 2021
current time after 30 sec: Mon Dec 20 20:42:17 2021
time t: time.struct_time(tm_year=2021, tm_mon=12, tm_mday=20, tm_hour=20, tm_min=41, tm_sec=47, tm_wday=0, tm_yday=354, tm_isdst=0)
current year: 2021
current month: 12
current day: 20
current hour: 20
current minute: 41
current second: 47
current weakday: 0
>>>
```

C) calendar module

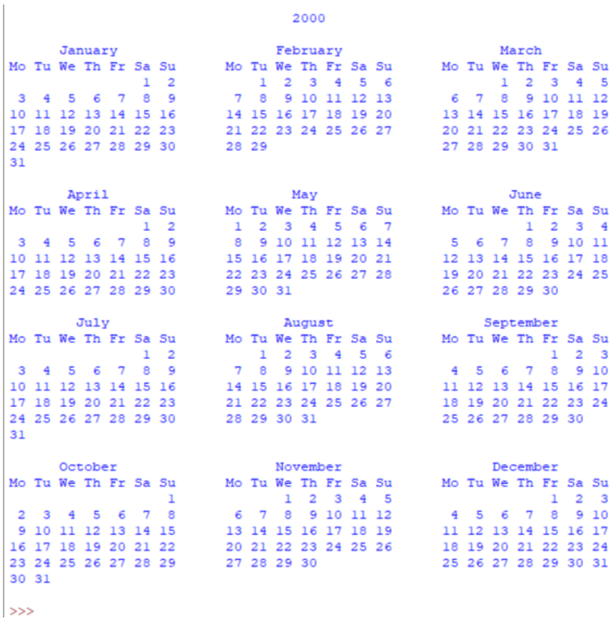
PROGRAM

```
import calendar

mm=int(input("Enter month:"))

yy=int(input("Enter year:"))

print(calendar.month(yy,mm))        #calendar of a given month

#print(calendar.calendar(2022))      #calendar of a given year

print(calendar.calendar(2000))
```

OUTPUT



```
>>> 
==================== RESTART: D:\MCA\python_lab\CO3\q3.py ====================
Enter month:12
Enter year:2020
   December 2020
Mo Tu We Th Fr Sa Su
    1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

                                  2000

      January                   February                   March
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2          1  2  3  4  5  6             1  2  3  4  5
 3  4  5  6  7  8  9       7  8  9 10 11 12 13       6  7  8  9 10 11 12
10 11 12 13 14 15 16      14 15 16 17 18 19 20      13 14 15 16 17 18 19
17 18 19 20 21 22 23      21 22 23 24 25 26 27      20 21 22 23 24 25 26
24 25 26 27 28 29 30      28 29                     27 28 29 30 31
31

       April                      May                       June
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2       1  2  3  4  5  6  7                1  2  3  4
 3  4  5  6  7  8  9       8  9 10 11 12 13 14       5  6  7  8  9 10 11
10 11 12 13 14 15 16      15 16 17 18 19 20 21      12 13 14 15 16 17 18
17 18 19 20 21 22 23      22 23 24 25 26 27 28      19 20 21 22 23 24 25
24 25 26 27 28 29 30      29 30 31                  26 27 28 29 30

        July                     August                  September
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2          1  2  3  4  5  6                   1  2  3
 3  4  5  6  7  8  9       7  8  9 10 11 12 13       4  5  6  7  8  9 10
10 11 12 13 14 15 16      14 15 16 17 18 19 20      11 12 13 14 15 16 17
```



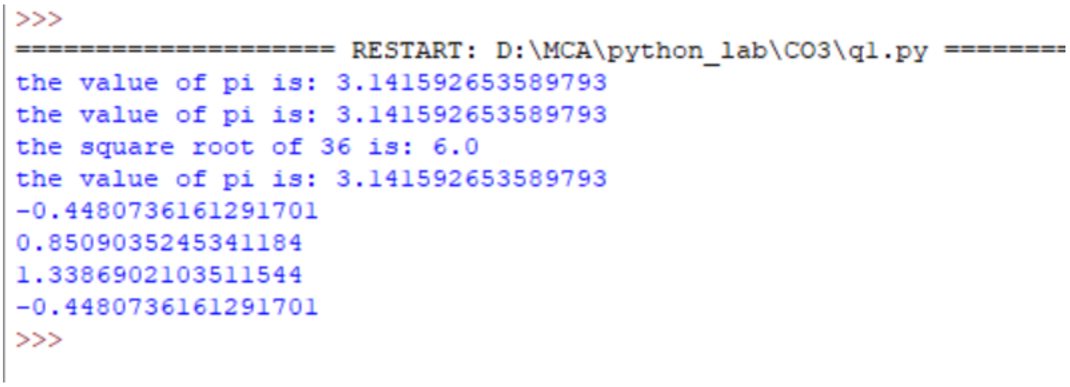
```
                                  2000

      January                   February                   March
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2          1  2  3  4  5  6             1  2  3  4  5
 3  4  5  6  7  8  9       7  8  9 10 11 12 13       6  7  8  9 10 11 12
10 11 12 13 14 15 16      14 15 16 17 18 19 20      13 14 15 16 17 18 19
17 18 19 20 21 22 23      21 22 23 24 25 26 27      20 21 22 23 24 25 26
24 25 26 27 28 29 30      28 29                     27 28 29 30 31
31

       April                      May                       June
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2       1  2  3  4  5  6  7                1  2  3  4
 3  4  5  6  7  8  9       8  9 10 11 12 13 14       5  6  7  8  9 10 11
10 11 12 13 14 15 16      15 16 17 18 19 20 21      12 13 14 15 16 17 18
17 18 19 20 21 22 23      22 23 24 25 26 27 28      19 20 21 22 23 24 25
24 25 26 27 28 29 30      29 30 31                  26 27 28 29 30

        July                     August                  September
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                1  2          1  2  3  4  5  6                   1  2  3
 3  4  5  6  7  8  9       7  8  9 10 11 12 13       4  5  6  7  8  9 10
10 11 12 13 14 15 16      14 15 16 17 18 19 20      11 12 13 14 15 16 17
17 18 19 20 21 22 23      21 22 23 24 25 26 27      18 19 20 21 22 23 24
24 25 26 27 28 29 30      28 29 30 31               25 26 27 28 29 30
31

      October                   November                  December
Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
                   1             1  2  3  4  5                   1  2  3
 2  3  4  5  6  7  8       6  7  8  9 10 11 12       4  5  6  7  8  9 10
 9 10 11 12 13 14 15      13 14 15 16 17 18 19      11 12 13 14 15 16 17
16 17 18 19 20 21 22      20 21 22 23 24 25 26      18 19 20 21 22 23 24
23 24 25 26 27 28 29      27 28 29 30               25 26 27 28 29 30 31
30 31

>>> 
```

D) math module

PROGRAM

```
import math
print("the value of pi is:",math.pi)


import math as m
print("the value of pi is:",m.pi)


from math import pi,sqrt
print("the square root of 36 is:",math.sqrt(36))
print("the value of pi is:",math.pi)


print(math.cos(90))
print(math.sin(45))
print(math.tan(180))
print(math.cos(90))
#print(math.cosec(90))
```

OUTPUT



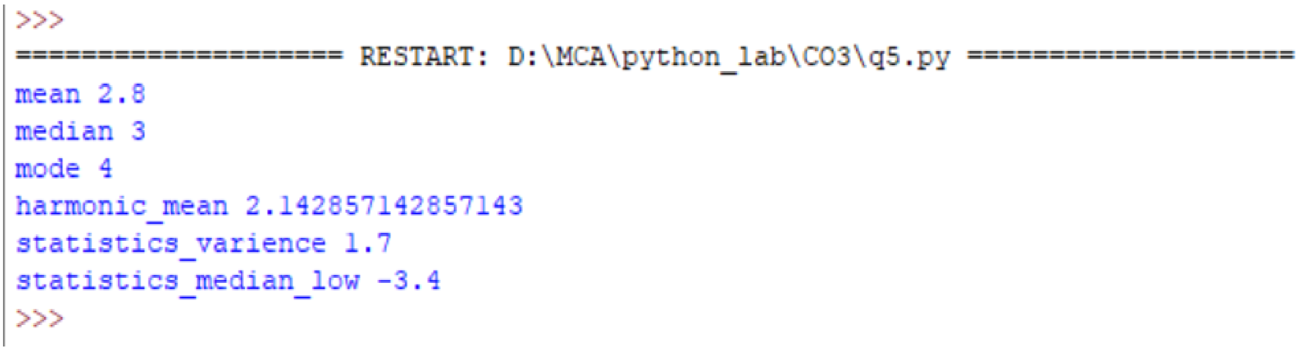
```
>>>
==================== RESTART: D:\MCA\python_lab\CO3\q1.py ========
the value of pi is: 3.141592653589793
the value of pi is: 3.141592653589793
the square root of 36 is: 6.0
the value of pi is: 3.141592653589793
-0.4480736161291701
0.8509035245341184
1.3386902103511544
-0.4480736161291701
>>>
```

E) Statistics module

PROGRAM

```
import statistics

l1=[1,2,3,4,4]

print("mean",statistics.mean(l1))

print("median",statistics.median(l1))

print("mode",statistics.mode(l1))

print("harmonic_mean",statistics.harmonic_mean(l1))

print("statistics_varience",statistics.variance(l1))

print("statistics_median_low",statistics.median_low([-11, 5.5, -3.4, 7.1, -9, 22]))
```

OUTPUT



```
>>>
==================== RESTART: D:\MCA\python_lab\CO3\q5.py ====================
mean 2.8
median 3
mode 4
harmonic_mean 2.142857142857143
statistics_varience 1.7
statistics_median_low -3.4
>>>
```

PROGRAM NO: 2 **DATE: 15-12-2021**

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)

PROGRAM

graphicsuse.py

```
from graphics import rectangle

from graphics import circle

from graphics.Dgraphics import  cuboid

from graphics.Dgraphics import sphere  #import cuboid and sphere


print("Read values: \n Rectangle:\n")

l=int(input("Enter length"))

b=int(input("Enter breadth"))

r_area=rectangle.area(l,b)

r_perimeter=rectangle.perimeter(l,b)


print("Circle:\n")

r=int(input("Enter radius"))

area=circle.area(circle.pi,r)

perimeter=circle.perimeter(circle.pi,r)

print("Read values: \n Cuboid:\n")

l=int(input("Enter length"))

b=int(input("Enter breadth"))

h=int(input("Enter height"))

c_area=cuboid.area(l,b,h)

c_perimeter=cuboid.perimeter(l,b,h)
```

```
print("Read values: \n Sphere:\n")
r=int(input("Enter radius"))
sphere_surf_area=sphere.surf_area(sphere.pi,r)
sphere_circumference=sphere.circumference(sphere.pi,r)
sphere_volume=sphere.sphere_volume(sphere.pi,r)

print("Area of Rectangle:",r_area)
print("Area of Circle:",area)
print("Perimeter of Rectangle:",r_perimeter)
print("Perimeter of Circle:",perimeter)

print("Area of cuboid:",c_area)
print("surface Area of sphere:",sphere_surf_area)
print("Perimeter of cuboid:",c_perimeter)
print("circumference of sphere:",sphere_circumference)
print("sphere_volume:",sphere_volume)
```

Package : graphics

circle.py

```
pi=3.14
def area(pi,r):
   return pi*r*r
def perimeter(pi,r):
   return 2*pi*r
```

---

rectangle.py

```
def area(a,b):
    return a*b
def perimeter(a,b):
    return 2*(a+b)
```

Sub-Package : ThreeD_graphics

cuboid.py

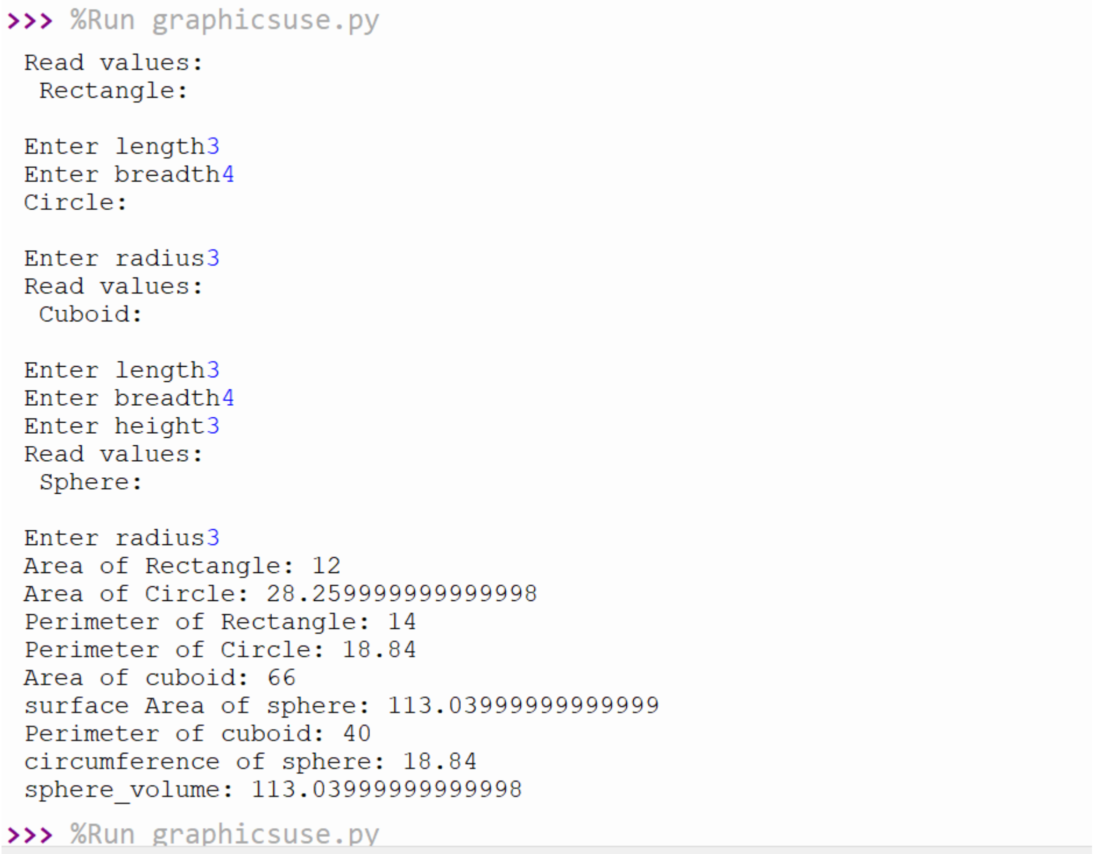
```
def area(l,b,h):
    return 2*(l*b+b*h+h*l)
def perimeter(l,b,h):
    return 4*(l+b+h)
```

----

sphere.py

```
pi=3.14
def sphere_volume(pi,r):
    return (4/3)*pi*r*r*r
def circumference(pi,r):
    return 2*pi*r
def surf_area(pi,r):
    return 4*pi*r*r
```

OUTPUT



```
>>> %Run graphicsuse.py
 Read values:
  Rectangle:

 Enter length3
 Enter breadth4
 Circle:

 Enter radius3
 Read values:
  Cuboid:

 Enter length3
 Enter breadth4
 Enter height3
 Read values:
  Sphere:

 Enter radius3
 Area of Rectangle: 12
 Area of Circle: 28.259999999999998
 Perimeter of Rectangle: 14
 Perimeter of Circle: 18.84
 Area of cuboid: 66
 surface Area of sphere: 113.03999999999999
 Perimeter of cuboid: 40
 circumference of sphere: 18.84
 sphere_volume: 113.03999999999998
>>> %Run graphicsuse.py
```

COURSE OUTCOME 4 (CO4)

PROGRAM NO: 1 **DATE: 03-01-2022**

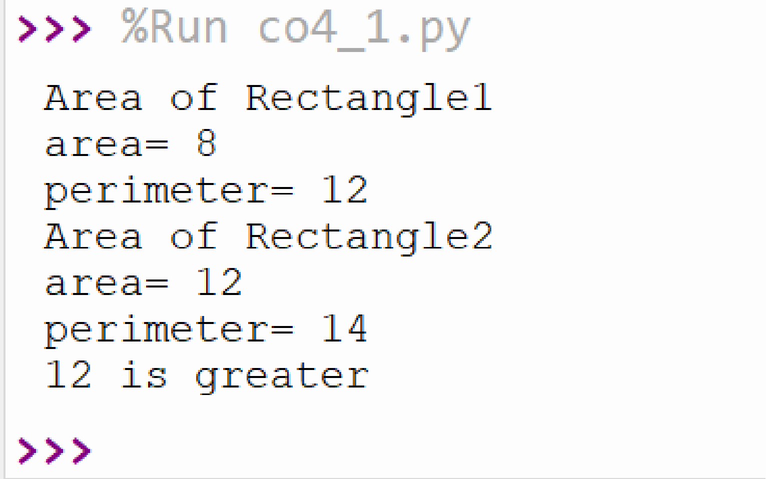
AIM : Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.

PROGRAM

```
class Rectangle:

   def __init__(self,length,breadth,ar):

      self.length=length

      self.breadth=breadth

      self.ar=0

   def area(self):

      self.ar=self.length*self.breadth

      return (self.ar)


   def perimeter(self):

      self.perimeter=2*(self.length+self.breadth)

      return (self.perimeter)


   def display(self):

      print("area=",self.ar)

      print("perimeter=",self.perimeter)


R1=Rectangle(2,4,0)

R2=Rectangle(3,4,0)


R1.area()

R1.perimeter()
```

```
R2.area()

R2.perimeter()

print("Area of Rectangle1")

R1.display()

print("Area of Rectangle2")

R2.display()


if (R1.ar>R2.ar):

    print(R1.ar,"is graeter")

else:

    print(R2.ar,"is greater")
```

OUTPUT



```
>>> %Run co4_1.py

 Area of Rectangle1
 area= 8
 perimeter= 12
 Area of Rectangle2
 area= 12
 perimeter= 14
 12 is greater

>>>
```

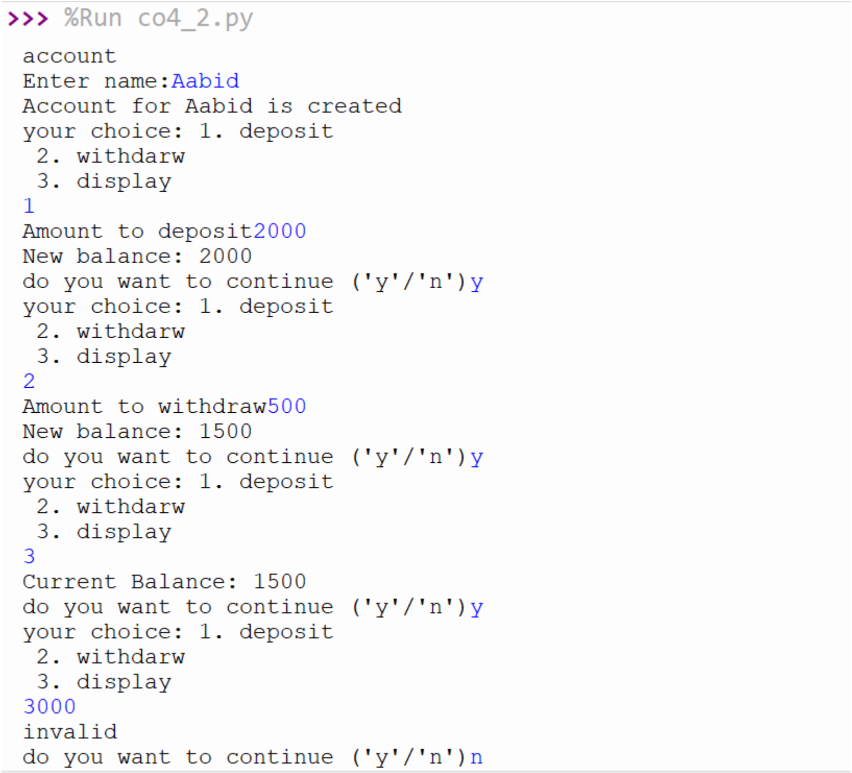
PROGRAM NO: 2 **DATE: 05-01-2022**

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.

PROGRAM

```
class Bank:
    def __init__(self,bal=0):
        #self.accno=accno
        #self.name=name
        #self.acctype=acctype
        self.bal=bal
        name=input("Enter name:")
        print("Account for",name,"is created")


  def deposit(self):
        amount=int(input("Amount to deposit"))
        self.bal=self.bal+amount


             print("New balance:",self.bal)
    def withdarw(self):
        amount=int(input("Amount to withdraw"))
        if(self.bal>amount):
            self.bal=self.bal-amount
            print("New balance:",self.bal)
        else:
            print("insufficient amount")
            print("balance:",self.bal)
    def display(self):

        print("Current Balance:",self.bal)

print("account")

b1=Bank()

opt='y'

while(opt=='y'):

    #print("your choice: 1. deposit \n 2. withdarw \n 3. display\n")

    choice=int(input("your choice: a. deposit \n 2. withdarw \n 3. display\n"))

    if(choice == 1):

        b1.deposit()

    elif(choice==2):

        b1.withdarw()

    elif(choice==3):

        b1.display()

    else:

        print("invalid")

     opt=input("do you want to continue ('y'/'n')")
```

OUTPUT



```
>>> %Run co4_2.py
 account
 Enter name:Aabid
 Account for Aabid is created
 your choice: 1. deposit
  2. withdarw
  3. display
 1
 Amount to deposit2000
 New balance: 2000
 do you want to continue ('y'/'n')y
 your choice: 1. deposit
  2. withdarw
  3. display
 2
 Amount to withdraw500
 New balance: 1500
 do you want to continue ('y'/'n')y
 your choice: 1. deposit
  2. withdarw
  3. display
 3
 Current Balance: 1500
 do you want to continue ('y'/'n')y
 your choice: 1. deposit
  2. withdarw
  3. display
 3000
 invalid
 do you want to continue ('y'/'n')n
```

PROGRAM NO: 3 **DATE: 05-01-2022**

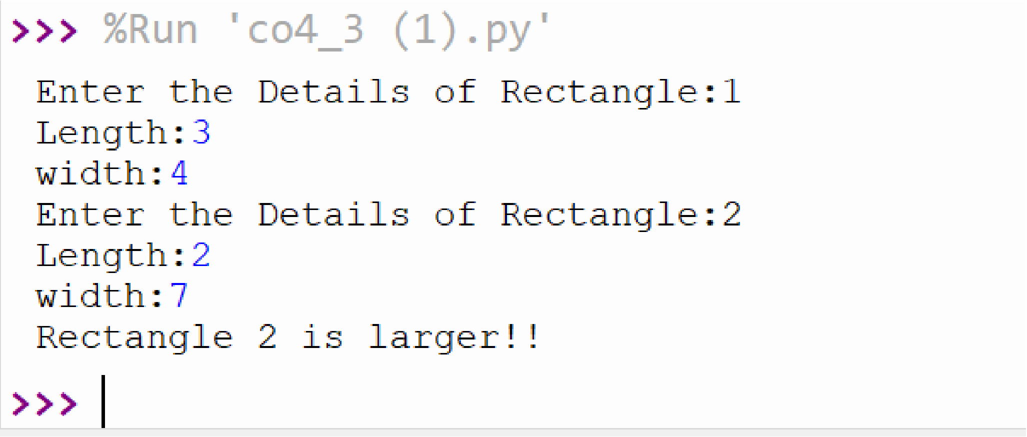
AIM : Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.

PROGRAM

```
class rectangle:
    def __init__(self,length,width):
        self.length=length
        self.width=width
    def __lt__(self,a1):
        area1=self.length*self.width
        area2=a1.length*a1.width
        if(area1>area2):

  return(True)
        else:
            return(False)

print("Enter the Details of Rectangle:1")
l1=int(input("Length:"))
w1=int(input("width:"))
r1=rectangle(l1,w1)
print("Enter the Details of Rectangle:2")
l2=int(input("Length:"))
w2=int(input("width:"))
r2=rectangle(l2,w2)
```

```
if(r1>r2):
    print("Rectangle 2 is larger!!")
else:
    print("Rectangle 1 is larger!!")
```

OUTPUT



```
>>> %Run 'co4_3 (1).py'
 Enter the Details of Rectangle:1
 Length:3
 width:4
 Enter the Details of Rectangle:2
 Length:2
 width:7
 Rectangle 2 is larger!!
>>> 
```

PROGRAM NO: 4 **DATE: 10-01-2022**

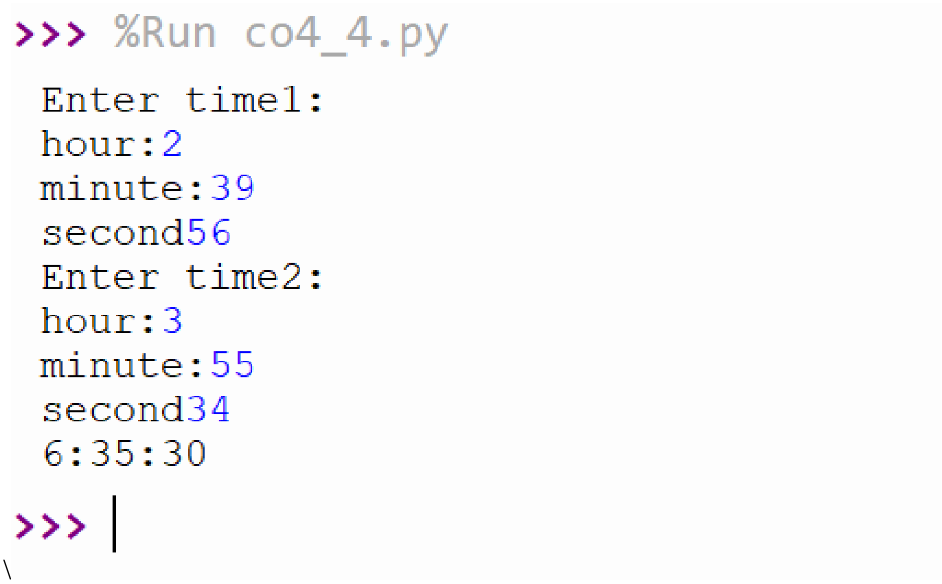
AIM : Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time

PROGRAM

```
class Time:
    def __init__(self,hour,minute,second):
        self.__hour=hour

self.__minute=minute
        self.__second=second
    def __add__(self,a2):
        second=self.__second+a2.__second
        minute=self.__minute+a2.__minute
        hour=self.__hour+a2.__hour
        if(second>60):
            second=second-60
            minute=minute+1
        if(minute>60):
            minute=minute-60
            hour=hour+1
        return hour,minute,second
print("Enter time1:")
h1=int(input("hour:"))
m1=int(input("minute:"))
s1=int(input("second"))
t1=Time(h1,m1,s1)
```

```
print("Enter time2:")

h2=int(input("hour:"))

m2=int(input("minute:"))

s2=int(input("second"))

t2=Time(h2,m2,s2)

hr,min,sec=t1+t2

print(hr,end=":")

print(min,end=":")

print(sec,end=" ")
```

OUTPUT



```
>>> %Run co4_4.py
 Enter time1:
 hour:2
 minute:39
 second56
 Enter time2:
 hour:3
 minute:55
 second34
 6:35:30

>>> 
```

\

PROGRAM NO: 5 **DATE: 10-01-2022**

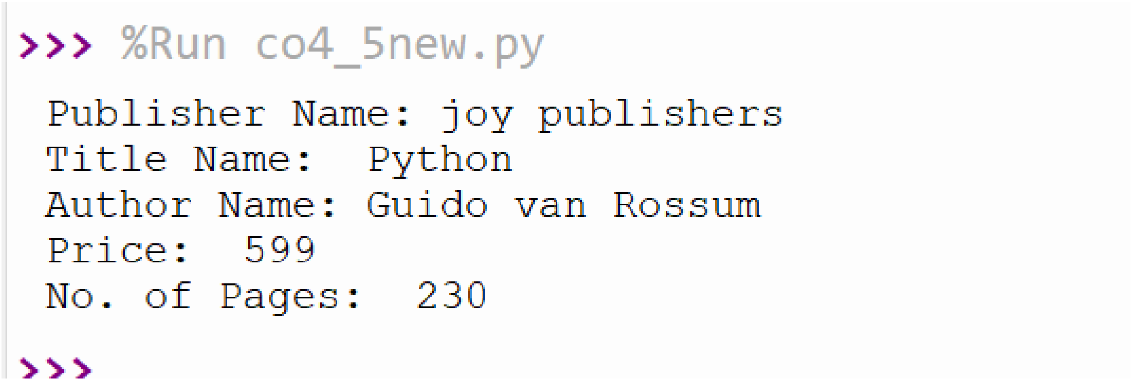
AIM : Create a class Publisher (name). Derive class Book from Publisher with attributes title and author. Derive class Python from Book with attributes price and no_of_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overriding.

PROGRAM

```
class publisher:
    def __init__ (self,pn):
        self.publishername=pn

    def display(self):
        print("Publisher Name:",self.publishername)

class book(publisher):
    def __init__ (self,pn,tt,aut):
        super(). __init__(pn)
        self.title=tt
        self.author=aut

    def display(self):
        super().display()
        print("Title Name: ",self.title)
        print("Author Name:",self.author)

sclass python(book):
    def __init__ (self,pn,tt,aut,pr,pg):
```

```
 super(). __init__(pn,tt,aut)
        self.price=pr
        self.page=pg

    def pythondisplay(self):
        print("Price: ",self.price)
        print("No. of Pages: ",self.page)



obj=python("joy publishers","Python","Guido van Rossum",599,230);
obj.display()
obj.pythondisplay();
```

OUTPUT



```
>>> %Run co4_5new.py
 Publisher Name: joy publishers
 Title Name:  Python
 Author Name: Guido van Rossum
 Price:  599
 No. of Pages:  230
>>>
```

COURSE OUTCOME 5 (CO5)

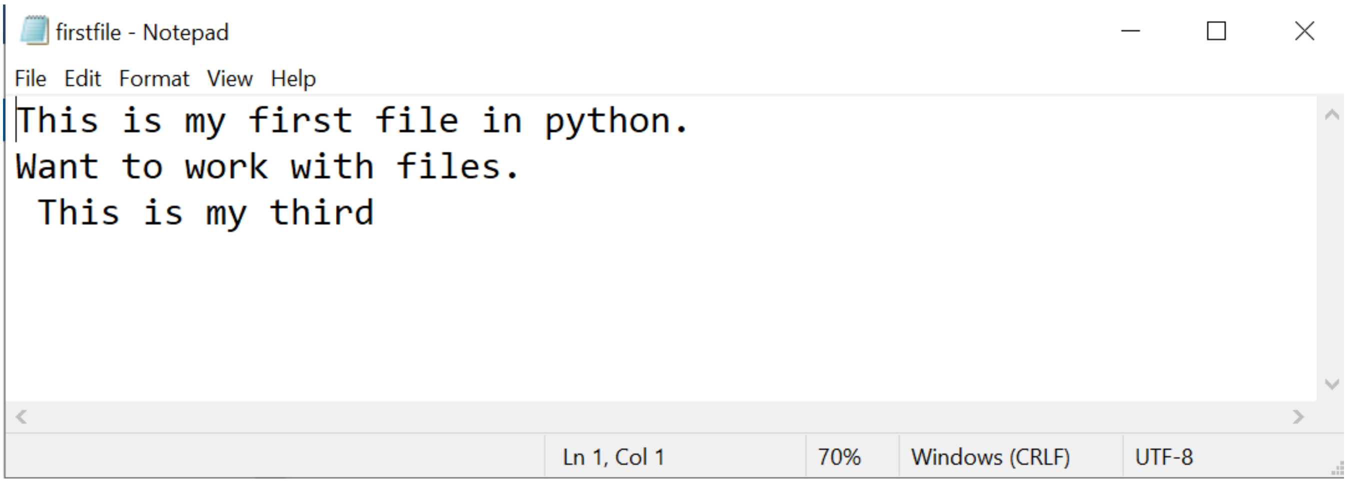
PROGRAM NO: 1 **DATE: 17-01-2022**

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

PROGRAM

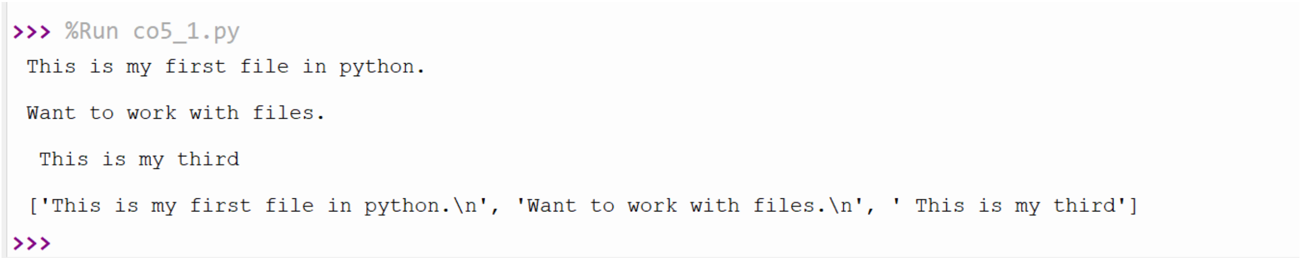
```
f1=open("firstfile.txt","w")
f1.write("This is my first file in python.\nWant to work with files.\n This is my third")
f1.close()
f1=open("firstfile.txt","r")
f1.seek(0,0)
ff=f1.readlines()
for x in range(0,len(ff)):
   print(ff[x])
print()
print(ff)
f1.close()
```

pythonfile.txt



```
This is my first file in python.
Want to work with files.
 This is my third
```

OUTPUT



```
>>> %Run co5_1.py
 This is my first file in python.

 Want to work with files.

  This is my third

 ['This is my first file in python.\n', 'Want to work with files.\n', ' This is my third']
>>>
```

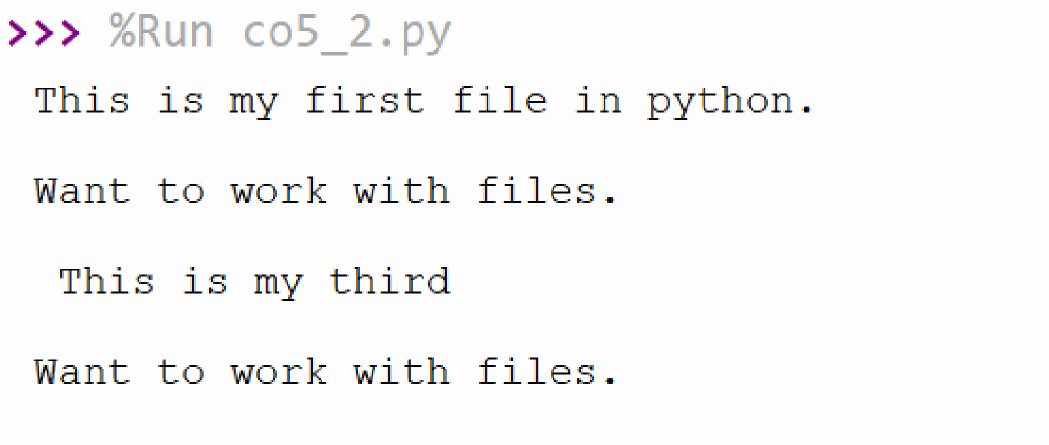
PROGRAM NO: 2 **DATE: 17-01-2022**

AIM : Write a program to copy odd lines of one file to other.

PROGRAM

```
f1=open("firstfile.txt","r")

for x in f1:

    print(x)

f1.seek(0,0)

print()

f2=open("odd.txt","w")

ff=f1.readlines()

with open('odd.txt','w') as f2:

     for x in range(0,len(ff)):

        if(x%2!=0):

            print(ff[x])

            f2.write(ff[x])
```

OUTPUT



```
>>> %Run co5_2.py
 This is my first file in python.

 Want to work with files.

  This is my third

 Want to work with files.
```

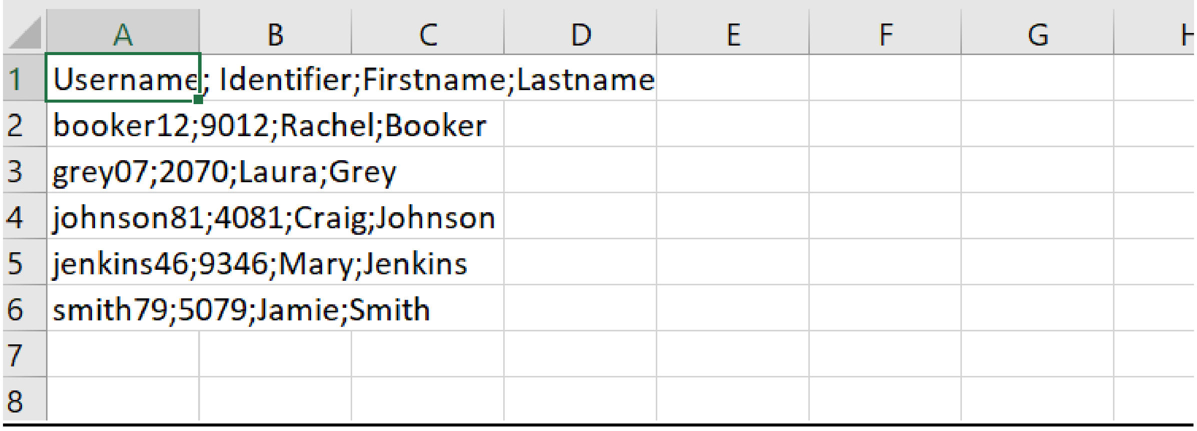
PROGRAM NO: 3 **DATE: 31-01-2022**

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

PROGRAM

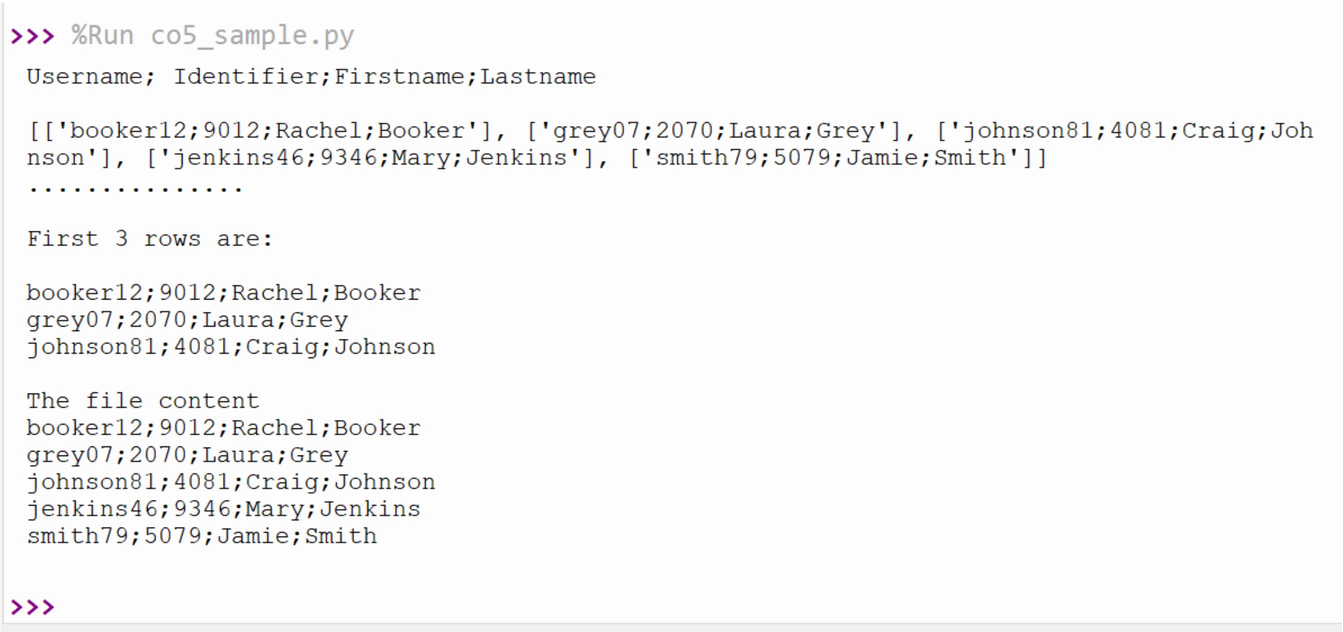
```
import csv
 # csv file name
filename = "username.csv"
 # initializing the titles and rows list
fields = []
rows = []
 # reading csv file
cf=open(filename, 'r') # creating a csv reader object
csvreader = csv.reader(cf)

 # extracting field names through first row
fields = next(cf)
print(fields)

 # extracting each data row one by one
for r in csvreader:
    rows.append(r)
 #print the list containing the rows of csv file
print(rows)
print("...............")
print('\nFirst 3 rows are:\n')
for r in rows[:3]:
    print(*r)
print()
print("The file content")
for sl in rows:
    for l in sl:
        print(l)
 #print(l,end=" ")
print()
cf.close()
```

username.csv



	A	B	C	D	E	F	G	H
1	Username; Identifier;Firstname;Lastname							
2	booker12;9012;Rachel;Booker							
3	grey07;2070;Laura;Grey							
4	johnson81;4081;Craig;Johnson							
5	jenkins46;9346;Mary;Jenkins							
6	smith79;5079;Jamie;Smith							
7								
8								

OUTPUT



```
>>> %Run co5_sample.py
 Username; Identifier;Firstname;Lastname

 [['booker12;9012;Rachel;Booker'], ['grey07;2070;Laura;Grey'], ['johnson81;4081;Craig;Joh
 nson'], ['jenkins46;9346;Mary;Jenkins'], ['smith79;5079;Jamie;Smith']]
 ...............

 First 3 rows are:

 booker12;9012;Rachel;Booker
 grey07;2070;Laura;Grey
 johnson81;4081;Craig;Johnson

 The file content
 booker12;9012;Rachel;Booker
 grey07;2070;Laura;Grey
 johnson81;4081;Craig;Johnson
 jenkins46;9346;Mary;Jenkins
 smith79;5079;Jamie;Smith


>>>
```

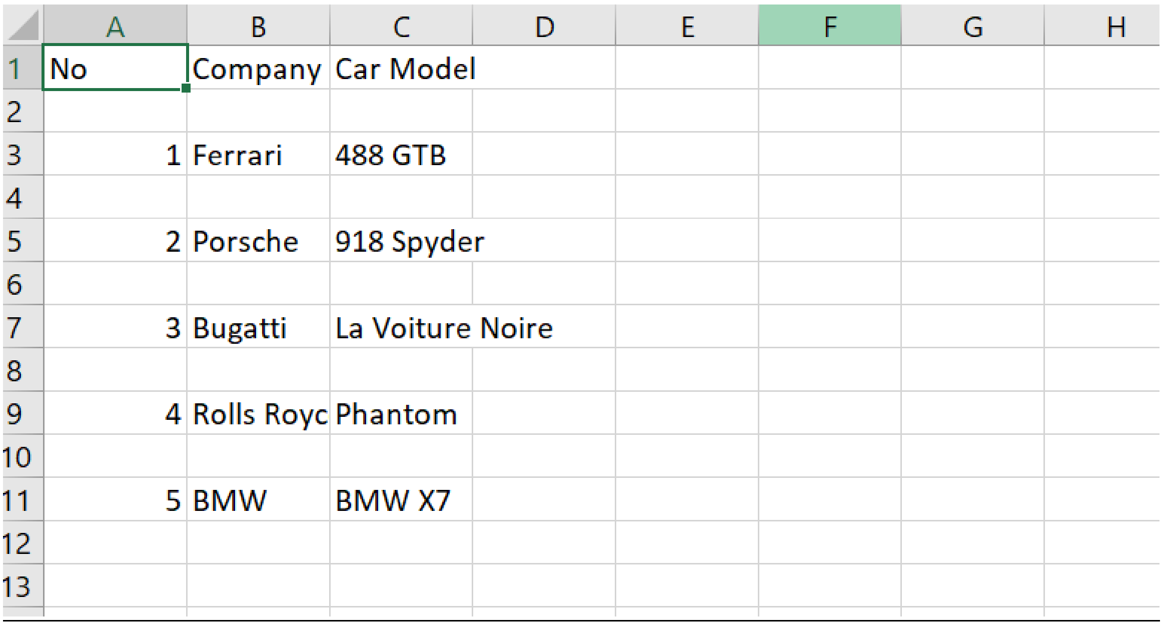
PROGRAM NO: 4 **DATE: 31-01-2022**

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

PROGRAM

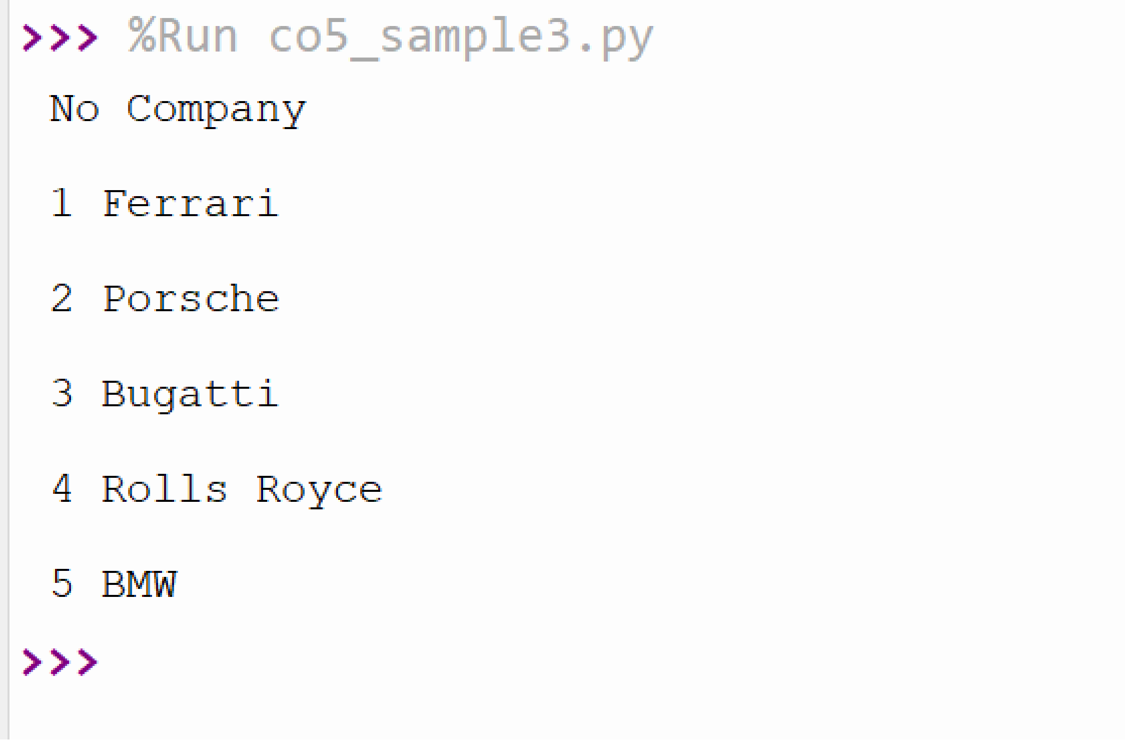
```
import csv
filename = "names.csv"
cf=open(filename, 'r')
 #csvreader = csv.reader(cf)
data = csv.DictReader(cf)
print("No Company")
for r in data:
    print(r['No'], r['Company'])
```

cardetails.csv



	A	B	C	D	E	F	G	H
1	No	Company	Car Model					
2								
3	1	Ferrari	488 GTB					
4								
5	2	Porsche	918 Spyder					
6								
7	3	Bugatti	La Voiture Noire					
8								
9	4	Rolls Royc	Phantom					
10								
11	5	BMW	BMW X7					
12								
13								

OUTPUT



```
>>> %Run co5_sample3.py
 No Company

 1 Ferrari

 2 Porsche

 3 Bugatti

 4 Rolls Royce

 5 BMW
>>>
```

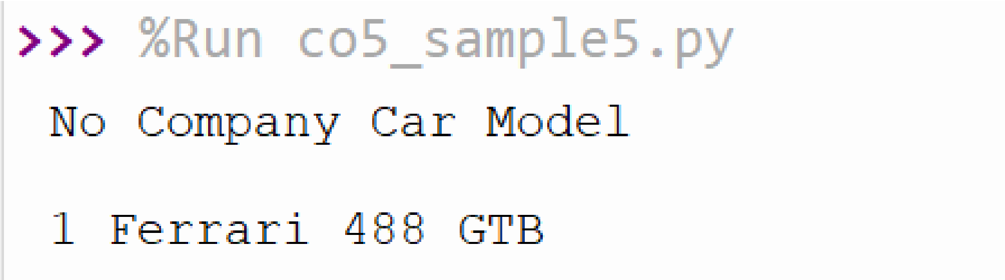
PROGRAM NO: 5 **DATE: 31-01-2022**

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.

PROGRAM

```
import csv
field_names = ['No', 'Company', 'Car Model']
cars = [
{'No': 1, 'Company': 'Ferrari', 'Car Model': '488 GTB'},
{'No': 2, 'Company': 'Porsche', 'Car Model': '918 Spyder'},
{'No': 3, 'Company': 'Bugatti', 'Car Model': 'La Voiture Noire'},
{'No': 4, 'Company': 'Rolls Royce', 'Car Model': 'Phantom'},
{'No': 5, 'Company': 'BMW', 'Car Model': 'BMW X7'},
]
with open('Names1.csv', 'w') as csvfile:
    writer = csv.DictWriter(csvfile, fieldnames = field_names)
    writer.writeheader()
    writer.writerows(cars)#print(".................")
filename = "names1.csv"
cf=open(filename, 'r')
rows=[]
csvreader = csv.reader(cf)
for r in csvreader:
    rows.append(r)
for r in rows[:3]:
    print(*r)
```

OUTPUT



```
>>> %Run co5_sample5.py
 No Company Car Model

 1 Ferrari 488 GTB
```