Program No:1

Aim : Python program to find area.

```
def area[r]:
    Pi = 3.14
    return = Pi*(r*r);
num= float(Input("enter the value for:")).
Print ("Area is %.6f" % area(num)).
```

Result :

The program has been executed and the output was verified.

Output·

```
Enter the value for : 3.

Area is 28.26000.
```

Program No : 2

Aim: Python program to find largest among 3 Numbers.

```
number 1 = float (Input ("enter the first number:")).
number 2 = float (Input ("enter the second number:")).
number 3 = float (Input ("enter the third number:")).
    If (number1> number 2) and (number1> number 3):
        Largest = number 1.
    elif (number 2 > number 1) and (number2> number 3):
        largest = number 2.
    else:
        largest = numbers
Print ("The largest number is", largest).
```

Result:

The program has been executed and the output was verified.

Output

```
enter the first number :2
enter the second number :4.
enter the third number : 5

the largest number is 5.
```

Progeam No :3.

Aim : Python progeam to find square of a numbee

```
digit = Int(Input ("entee an ibtegee numbee: "))
square = digit * digit
Printf("square of {digit} is {square}").
```

Result :

The progeam has been executeel and the output was veeified.

Output

```
Enter an Integer number :4.
Square of 4 is 16.
```

Program No:4

Aim : Python program to find area of Circle .

```
From math Import pi.
r = float (Input the radius of the circle:"))
Print ("the area of the circle with radius "
        + str(r)+ " is:" + str (pi * r**2))
```

Result:

The program has been executed and the output was verified

Output

```
Input the radius of Circle:4.
The area of the circle with radius 4.0 is
50.2654.
```

Program No: 5

Aim: Python program to find square of n.

```
list 1 = [14, 20,13, 8,6,2].

for n in List 1:

        Square = n**n.

  Print (n, Squared is', square).
```

Result.

The program has been executed and the output was verified.

Output

```
14 Squared is 196.
20 squared is 400
13 Squared is 169.
8 Squared is 64
6 Squared is 36
2 squared is 4.
```

Program :6

Aim: Python program to find vowels in a string

```
string A = "Hello .. How are you"
Print ("Given string :\n", string A).

Vowels = "AaEeIiOoUu".
res = Set([each for each in string A if each in Vowels]).
Print ("The vowels present in the string :\n", res).
```

Result

The program has been executed and the output was verified

Output

```
Given String:
        Hello... how are you.
The vowels in the string:
    {'u', 'a', 'e', 'o'}.
```

Program 7

Aim : Python program to count words in a sentence

```
def word = Count (str):.
    Counts = dict ()
    Words = str . split ()
for word in words:
    If word is counts:
        Counts [Word]+= 1
    else :
      Counts [word] = 1.

  Retuen Counts.
Print (Word- count ('When you change the quality of youe
thinking, you change the quality of youe life
sometimes Instantly')).
```

Result:

The program has been executed and the output was veeified.

Output

```
{'when':1 , 'you':2, 'Change':2, 'the':2, 'quality':2,
 'of':2, 'your':2, 'thinking':1, 'life':1, 'Sometimes':1,
 'Instantly':1}.
```

Program No: 8.

Aim: Python program to count a in a list.

```
a = ['Anto', 'Rehan', 'Billan', 'Donis'].

str1 = (''. join(a)).

Count = 0.

for i in str1:
    if i = 'a':
        Count = Count + 1.

Print("Count of a in the list is:"
            + str (count)).
```

Result:

The program has been executed and the output was verified.

Output

Count of a in the list is : 3.

Progeam No:9.

Aim: Python progeam to check the length of lists.

```
List 1 = [10,10,11,12,12,13,14,16,15,16,12].
list 2 = [16,12,13,14,15,16,10,11, 12,10,12].

 len 1 = len(list 2).
 len 2 = len (list 2)
  If len 1 = len2:
   Print ('both list have equal length').
else:
   Print ('both list doesn't have equal length')
```

Result

The progeam has been executed and the output was veeified.

Output:

Both list have equal length.

Program No: 10.

Aim: Python program to check the sum of lists.

```
list 1 = [10,10,11,12,12, 13, 14, 16, 15, 16, 12].
list 2 = [16, 12, 13, 14, 15 , 16, 10, 11 ,12, 10, 12].

total 1 = sum (list 1).
total 2 = sum (list 2).
If total 1 = total 2 :
 Print ('both list have equal sum').
 else:
 Print ('both list doesn't have equal sum').
```

Result

The program has been executed and the output was verified.

Output

Both list have equal sum.

Program No: 11

Aim: Python program to check the common elements in the lists.

```
list 1 = [10,10,11,12,12,13, 14, 16, 15, 16, 12]
list 2 = [10,10,11,12,12, 16, 14, 16, 15, 19, 12].

for value in list 1:
        If value in list 2:
                Common = 1.

    If common = 1:
        Point ("there are Common elements").
    else: print ("no common elements").
```

Result:

The program has been executed and the output was verified.

Output

There are common elements.

Program No:12.

Aim: Python program to replace a character.

```
def Change-char (str 1):
     char = str 1 [0].
    str 1 = str 1 . replace( char, '$').
    str 1 = Char + str 1 [1:].
 Print (change-char ('refresh')).
```

Result

The program has been executed and the output was verified

Output

ref $esh

Program No : 13.

Aim: Python program to exchange the first and last letter in string

```
def change_string (str 1):
        return str 1[-1:] + str 1[1:-1] + str 1[:1].
Print (change_string ('pineapple')).
```

Result

The program has been executed and output was verified.

Output:

```
eineapplp
```

Program No: 14.

Aim: Python program to merge 2 dictionaries

```
def merge (dict1:dict2):
        return (dict2 .update (dict1)).

dict 1 = {'a': 10, 'b':8}.
dict 2 = {'d':5, 'c':2}
Print (Merge (dict1, dict2)).
Print (dict2).
```

Result :

The program has been executed and the outp

was verified.

Output

```
None
{'d':5, 'c':2, 'a':10, 'b':6}.
```

Program :15

Aim : Python program to ascend desend dictonary

```
Import Operator
    d = {1:2, 3:4, 4:3, 2:1, 0:0}.
Print ('Original dictonary :', d).
Sorted -d = Sorted (d.items (), Key = Operator . itemgetter (1)).
Print ('Dictonary in ascending order by value :',
        Sorted -d ).
Sorted-d = dict(Sorted (d.items (), Key = Operator. Itemgetter (1),
                reverse = True)).
Print ('Dictonary in ascending order by value : ', Sorted-d)
```

Result

The program has been executed and the output was verified.

Output

```
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 16

Aim : Python program to remove even numbers from the list.

```
list = [11,22, 33,44, 55,66, 77,88,99].

 Print (list).

 for i in list :
      if (i%2==0)
          list remove (i)

Print ("list after removing:", list).
```

Result

The program has been executed and the output was verified.

Output

```
[11, 22, 33, 44, 55, 66, 77, 88, 99].

list after removing: [11, 33, 55, 77, 99].
```

Program No : 17

Aim: Python progcamme to find gcd of numbee.

```
def gcd (a,b) :
  if (b==0)
   return a.
   return gcd (b, a % b).

 a = 45
 b = 65
 if (gcd (a,b)) :
  Print ('gcd of ', a, 'and ', b ,'is', gcd (a,b)).

else :
   Print ('not found')
```

Result

The progeam has been executed and the output was veeified

Output:

```
GCD of 45 and 65 is 5.
```

Progeam No:18·

Aim: Python progeam to find factorial of a numbee.

```
num = int (Input ("entee a numbee:")).
factorial = 1
if num < 0 :
  Print ("Sorry, factorial does not exit for negative
        numbees").
elif num = 0:
      print ("the factorial of 0 is 1"),
else:
    for 1 in range (1, num+1):
      factorial = factorial * i
 Print (" the factorial of ", num, " is ", factorial)
```

Result.

The progeam has been executed and ~~ve~~ the output was veeified.

Output

```
Enter a number: 5
The factorial of 5 is 120.
```

Progeam No :19.

Aim: Python progeam to find fibonnaci sequence.

```
def recure-fibo (n):
    if n <= 1 :
    retuen n
 else :
    retuen (recure-fibo(n-1) + recure-fibo (n-2))

nteems = Int (Input ("How many teems ?"))
  If n teems <= 0:
   Print ("please entee a positive integee.").

else!
   Print (" Fibonacci sequence:").
   for i in range (n teems):
    Print (recure-fibo (i)).
```

Result

The progeam has been executed and the output was veeified.

Output.

```
How many terms ?. 4.
Fibonacci sequence :
   0
   1
   1
   2.
```

Program No: 20

Aim: Python program to perform ~~using~~ string function.

```
def add string (str 1):
      length = len (str 1)
    If length > 1 :
        if str 1 [-3:] = "ing" :
        str 1 += 'ly'
     else :
          str 1 += ing'
    Return str 1
  Print (add-string ('do')).
  Print (add - string ('according'))
```

Result

The program has been executed and the output was verified.

Output.

doing
accordingly

Program No: 21

Aim: Python program to perform the sums of given items.

```
numbers = [1, 2, 3, 4, 5, 2, 5].
Sums = Sums [numbers].
Print (Sums).
```

Result

The program has been executed and the output was verified.

Output

22.

Program No: 22

Aim: Python program to find perfect even square numbers in a range.

```
num1 = int(input("Enter a number :"))
num2 = int(input("Enter a number :"))
for i in range (num1, num2+1):
    for j in range (32,100+1):
        if i = j*j:
    string = str(i).
    if int (string [0]) % 2 == 0 and.
    int (string [1]) % 2 == 0 and.
    int (string [2] % 2 == 0 and.
    int (string [3] % 2 == 0 ~~and~~.
    print (i)
```

Result

The program has been ~~ve~~executed and the output was verified.

Output

```
Enter a number :4444.
Enter a number : 9999.

4624
6084
6400
8464.
```

Program No : 23.

Aim: Python programme to display the given pyramid with step number accepted from user.

```
lines = int (Input("Enter a number:")).

 ~~Of~~ ~~Ex~~  i = 1
                j = 1.
 while  i < = lines
            j = 1
        while j < = i :

    temp = i * j

Print (temp ,end = ' ' , flush = true).
Print (" " , end = ' ' , flush = true).

   ~~P~~ j = j + 1. ;
      Print (" ") ;
     i = i + 1;
```

Result

The program has been executed and the out put was verified.

Output

```
Entee a cnumbee:4.

1
2 4
3 6 9
4 8 12 16.
```

Program. 24

Aim: Python programme to count the number of characters in a string.

```
def char - frequency (str 1):
  dict = {}.
  for n in str1:
    keys = dict.keys ()
    if n in keys:
     dict [n]+= 1
   else:
   dict[n] = 1
 return dict
Print (char-frequency ('hello how are you')).
```

Result:

The program has been executed and the output was verified.

Output

```
{'h':2,'e':2,'l':2,'o':3,'':3,'w':1,'a':1,
 's':1,'y':1,'v':1}.
```

Program No : 25

Aim : Python programme to accept a list of words, and returns lengths of longest word.

```
def find(word) :
    w1 = []
  for n in words:
     w1 . append ((len(n), n))
  w1 . sort()
  result = w1[-1][0], w1[-1][1]
  Print ("Longest Word :", result [1]).
  Print ("Lengths of the longest word :", result [0])
  find ["hello", "morning", "hi"]
```

Result

The program has been executed and output was verified.

Output

```
longest word: morning
length of the longest word: 7.
```

Program No: 06

Aim: Python program to construct pattern using nested loop.

```
def star ():
    n=5.
    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 (" ").
 star ()
```

Result.

The program has been executed and the output was verified.

Output

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

Progeam No: 27

Aim: Python progeam to print factors of a Numbee.

```
def print - factors (x):
  Print (" the factor of," x, " are :")
    for i in range (1, x+1):
        if x % i = = 0
        Print (i)

    Print - factor s(032).
```

Result

The progeam has been excuted and the output was veeified.

Output

```
The factors of 232 are:

1
2
4
8
29.
58
116
232
```

Program No: 28.

Aim: Python programe to write lambda functions to find area of square, triangle, rectangle.

```
Print ("Enter the length of side of square:")
s = int (Input ("Enter your value:"))
Print ("Enter the length and breadth of rectangle")
l = Int (Input ("Enter your value:"))
b = int (input ("Enter your value:")).
Print ('Enter the base and height of triangle')
h = int (input ("Enter your value:"))
d = int (Input ("Enter your value:")).
x = lambda s: s*s
y = lambda l,b: l*b

t = 0.5
z = lambda h,d,t: h*d*t.
Print ("Area of square is:", x(s))
Print ("Area of rectangle is:", y (l,b))
Print ("Area of triangle is:", z (h,d,t)).
```

Result

The program has been executed and the output was verified

Output

```
Enter the lengths of a side of square
Enter your value :2
Enter the lengths and breadths of rectangle
Enter your value :4.
Enter your value :2.
Enter the base and height of triangle
Enter your value : 3
Enter your value :2
Area of square : 4
Area of rectangle : 8
Area of triangle : 30
```

Program No: 29

Aim: Python program to display future leapyears from current year to a final year entered by user.

```
import datetime.
a = datetime  datetime.now()
a = int (a.year).
b = int (input("Enter final year:"))
Print ("\n leapyears:")
 for i in range (a,b,+1):
    if (i % 4 = = 0)
    Print (i)
```

Result

The program has been executed and the output was verified.

Output:

```
Enter final year : 2040.
leap years :

2024
2028
2032
2036
2040
```

Program No: 30

Aim : Python program to generate positive list of numbers from a given list of integers.

```
list 1 = [1, -1, 2, -5, 9, -2, -54, 87, -33, -46, 24, -67].
Pos = list ~~(x)~~ ()
 for i in list 1:
    if i > 0
    Pos . append (i)
Print('original list': list1).
Print('positive integer list :', pos).
```

Result

The program has been executed, the output was verified.

Output

Original list : [1, -1, 2, -5, 9, -2, -54, 87, -33, -76, 24, -6].

Positive Intgee list : [1, 2, 9, 87, 24].

Program No 31

Aim: Python program to find biggest of 3 numbers entered.

```
a = int(input ('Enter 1st no:'))
b = int(input ('Enter 2nd no:'))
c = int (input ('Enter 3rd no:'))
 if a>b and b>c:
   Print (a,'is the biggest number'):
  if b>a and b>c:
    Print (b,' is the biggest number').
  else:
    Print (c,'is the biggest number).
```

Result

The program has been executed and the output was verified.

Output

```
Entee 1st no:5
Entee 2nd no :6
Entee 3rd no : 8

8 is the highest numbee
```

Program No : 32

Aim! Python programs to create a list of colors from commas – separated color names, entered by user Display first and last colors.

```
Colors = (input ('Enter colors separated by commas:')).
        split (',').

Print ('First color :', Colors [0]).
Print ('Last color:', colors [len colors]-1]
```

Result.

The program has been executed and the output was verified.

Output

```
Enter colors separated by commas : Red, black,
yellow.

first color : red
last color : yellow
```

Program No : 33

Aim: Python program to print out all colors from colors list 1 not contained in color list 2

```
Colors 1= set((input('Enter colors separated by
            commas :')).split(',')).

Colors 2= set((input('Enter colors separated by
            commas :')).split(',')).

Print('Colors in color-list 1 not contained in
        color-list 2 are : ', list (Colors 1 - difference
            (Colors 2))).
```

Result

The program has been executed and the output was verified.

Output

```
Enter colors separated by commas: red,yellow,brown
Enter colors separated by commas: black,white
Colors in color-list 1 not contained in color list 2 are:
['brown', 'red', 'yellow']
```

Program No:34

Aim: Python program to create a package graphics with modulas rectangle, circle and subpackage 3D graphics with modules, cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that find area and perimeter of figures by different importing statements.

Circle.Py

```
def area(r):
    Print('Area of circle with radius', r, 'is:',
'%.2f'%(3.14*r*r), 'sq units').
def circumference(r):
    Print('Circumference of circle with radius',
    r, 'is:', '%.2f'%(3.14*2*r), 'units').
```

Rectangle.Py

```
def area(a,b):
    Print('Area of rectangle with sides', a, 'and', b,
    'is :', '%.2f'%(a*b), 'sq.units').
def perimeter(a,b):
    Print('Perimeter of rectangle with sides', a, 'and'
    ,b, 'is:', '%.2f'%(2*(a+b)), 'units').
```

Sphere.Py

```
def area(r):
Print('Area of sphere with radius', r, 'is:', '%
.2f'%(4*(3.14*r*r)), 'sq. units').
```

```
def peeimetee(r):
   Print ('peeimetee of (great circle of ) sphere with
   radius', r, 'is:', '%.2f'% (2*3.14*r), 'units').
```

Cuboid. Py.

```
def area(l,b,h):
   Print ('total surface area of cuboid with dimensions
   ;' l,',',b, ',',h,'is:', '%.2f'% (2*((l*b)}+(b*h)+(l*h))
   ,'sq.units').
def peeimetee.(l,b,h):
   Print ('peeimetee of cuboid with dimensions;
   l,',',b,',',h,'is:',' %.2f'% (4* (l+b+h)).,'units').
```

find. Peeimetee. Py

```
Import circle
from rectangle import*
from graphics. 3D-graphics import cuboid, sphere.
a= float(input ('Entee length of the rectangle :'))
b= float (input ('Entee breadth of the rectangle :')).
Peeimetee (a,b)
r= float(input('Entee the radius of the circle:'))
Circle . Circumference (r)
l = float (input ('Entee length of the cuboid:'))
b= float( input ('Entee breadth of the cuboid :'))
h= float( input('Entee height of the cuboid:'))
Cuboid. Peeimetee (l,b,h)
r= float(input('Entee the radius of the sphere :'))
Sphere. peeimetee (r).
```

find area py

```
Import circle
from rectangle import *
from graphics - 3D-graphics import cuboid, sphere.
a= float (input ('Enter length of the rectangle:'))
b= float (input ('Enter breadths of the rectangle:'))
area (a.b)
r= float (input ("Enter the radius of the circle:'))
circle = area (r)
l= float (input ('Enter lengths of cuboid:'))
b= float (input ('Enter breadths of cuboid:'))
h= float (input ('Enter height of Cuboid:'))
Cuboid= area(l,b,h)
 r= float (Input ('Enter the radius of sphere:'))
 Sphere. area (r)
```

Result

The program has been executed and the output w verified.

Output

```
Enter length of rectangle :4
Enter breadth of rectangle :3.
Perimeter of rectangle with sides 4.0 and 3.0 is:14.00 u
Enter the radius of the circle:2
Circumference of the circle with radius 2.0 is 12.56 uni
Enter length of the cuboid:5
Enter breadth of the cuboid:4
Enter height of the cuboid:3
Perimeter of cuboid with dimensions 5.0, 4.0, 3.0
is 48.00units.

Enter the radius of sphere:2
Perimeter of (great circle of) sphere with radius 2.0 is
12.56 units.

Enter length of rectangle= 2.
Enter breadth of rectangle = 3.
Area of rectangle with sides 2.0 and 3.0 is :6.00 Sq uni
Enter the radius of circle:4
Area of circle with radius 4.0 is 50.024 Sq units.
Enter length of cuboid:4
Enter breadth of cuboid:7
Enter height of cuboid : 2
Total surface area of cuboid with dimensions
4.0, 7.0, 2.0 is 100.00 Sq.units.
```

```
Enter the radius of sphere: 1
Area of sphere with radius 1.0 is 12.56 sq. units.
```

Program No: 35

Aim: Python program to recta create rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two rectangle objects by their area.

```
Class Rectangle
  def __init__(self, l, b):
    self.length = l.
    self.breadth = b
def area(self):
    return self.length * self.breadth
def perimeter (self):
    return 2 * (self.length + self breadth)
def imp (self, obj):
   if self.area() > obj.area():
   Print ('Rectangle with length =', self.length, 'and breadth =
   self.breadth, 'has the greater area').
elif self.area() < obj.area():
 Print ('Rectangle with length =', obj.length, 'and breadth =
 obj.breadth, 'has the greater area')
else:
Print ('they have equal area:')
r1 . Rectangle (9,3)
r2 . Rectangle (3,4)
r1 . imp(r2)
```

Result

The program has been executed and the output was verified.

Output

Rectangle with length = 9 and breadth = 3 has greater area.

Progeams No : 36

Aim: Python programs to 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.

```
Class Bank account:
   def — init — (self, a,n,t,b):
   self .acno= a.
  self - name = n
  self · type = t
  self . bal = b.
 def ·deposit (self,a):
      self·bal + =a.
 Print ('Rs.', a,' deposited! current balance is Rs.', self. bal).
def withdraw (self,a):
if self ·bal>=a:
  self . bal -=a
Print ('Rs .',a, 'withdeawn! current balance is Rs .', self.bal))
else!
Print ('Insufficient balance to make this transaction!')
a = int(input ('Entee account numbee:'))
n = int(input ('Entee name of the account holdee:')
t = input ('Entee account type: ').
b = float (Input ('Entee your balance:'))
ac 1 = Bank Account (a,n,t,b)
ac 1 · deposit (float (input '('Entee amount to deposit:'))
ac 1 .withdeaw (float (input'(Entee amount to withdeaw:'))
```

Result

The progeams has been executed and the output was verified

Output

```
Entee account numbee. ':00900909090909.
Enter name of the account holdee ':John.
Enter account type : Savings
Entee your balance '. 100000
Entee amount to deposit '. 300000
Rs. 300000.0 deposited ! Current balance is Rs. 400000.0
Entee amount to withdraw: 5000.
Rs 5000.0 withdrawn!. Current balance is ': Rs. 395000.0
```

Progeam No: 37

Aim: Python program to create rectangle class with attributes length and breadth and methods to find area and perimeter, compare 2 Rectangle objects by their area.

```
Class Rectangle:
     def __init__(self, l,b):
     self.length = l
     self.breadth = b
     def area(self):
      return self.length* self.breadth
     def perimeter(self):
      return 2 * (self.length+self.breadth)
  def cmp(self,obj):
   if self.area()>obj.area():
   Print('Rectangle with length=', self.length, 'and
   breadth=', self.breadth, 'has the greater area')
  elif self.area()<obj.area():
   Print('Rectange with length=', obj.length, 'and
   breadth=', obj.breadth, 'has the greater area').
  else: Print('They have equal area!')
   r1 = Rectangle(9,3)
   r2 = Rectangle(3,4)
   r1 = cmp(r2)
```

Result.

The program has been executed and the output was verified.

Output

Rectangle with length =9 and breadth =3 has the greatee area.

Program No: 38

Aim: Python program to create a class Rectangle with Private attribute length and width. Overload '<' operator to compare the area of 2 rectangles.

```
Class Rectangle:
  def _init_ (self,l,w):
  self._length = l
  self._width = w
  self.area = self_width * self._length
  def _lt_ (self. other):
If self.area < other.area:
 Print('Rectangle with length=', self._length,'
and width =', self._width, 'has the lesser area!')
elif other.area < self.area:
 Print('Rectangle with length=', other_length, 'and
 width=', other_width,' has the lesser area!")
else:
      Print('they have equal area!')
l= float(input('Enter length of 1st rectangle:'))
w: float(input('Enter width of 1st rectangle:'))
R1 = Rectangle(l,w)
l= float(input('Enter length of 2nd rectangle:'))
W= float(input('Enter Width of 2nd rectangle:'))
R2= Rectangle(l,w)
```

R1 < R2.

Result.

The program has been executed and the output was verified.

Output

```
Enter length of 1st rectangle : 7
Enter width of 1st rectangle : 8
Enter length of 2nd rectangle : 8
Enter width of 2nd rectangle : 7.
They have equal area!
```

Program No: 39.

Aim: Python program to 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.

```
Class Publisher:
    def __init__(self, name1):
    Self.name = name1
    def show(self):
   Pass
class Book(Publisher):
   def __init__(self, title1, author1, name1):
 Self title = title 1.
 Self. author = author 1
 Publisher __init__(sell, name 1).
 def show(self):
Pass.
Class python(Book):
  def __init__(Self, P, no, title1, author 1, name 1):
  Self. price = P.
  Self. no_of_pages = no.
 Book. __init__(Self, title 1, author 1, name 1)
```

```
def show (self):
    Print('Book title:', self.title).
    Print('Author:', self.author).
    Print('Publisher:', self.name).
    Print('Price:', self.price)
    Print('No of pages:', self.no_of_pages).

P1 = Python(565.90, 250, 'Programming with python,'
        'GV Rossum', 'ABC Books').
P1.show().
```

Result

The program has been executed and output was verified.

Output

```
Book title : Programming with Python.
Author ! GV Rossum
Publisher : ABC Books.
Price : 565.9.
No of pages : 850
```

Program No: 40

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

```
de file _ read (fname):
    with open(fname) as f:
 # # Content -list is the list that contains the
 read lines.
c = f .read lines ()
  Print (c)
# print (lens (c))
file- read ("demo.txt")
```

Result

The program has been executed and the output was verified.

Output

```
['A trailee is a vehicle designed for carrying bulk
mateeial, often on building states.\n', 'they are
distinguished from dump drucks by configuraton :
a dumped']
```

Program No:41

Aim: Python program to copy add lines of one file to other.

```
a = open ('demo.txt', 'r')
b = open('f.txt', 'w')
c = a.readlines()
for i in range (0, len(c)):
      if(i % 2 != 0):
      b.write(c[i])
     else:
      Pass
b.close()
b = open ('f.txt', 'r')
d = b.read()
Print (d)
a.close()
b.close()
```

Result.

The program has been executed and the output was verified.

Output

They are distinguished from dump trucks by configuration a dumper is usually an open 4-wheeled vehicles with the load skip in front of the driver.

Progeam No:42

Aim: Python progeam to read each raw from a given csv file and print a list of strings.

```
import csv
with open('temp.csv', newline='') as csvfile:
d= csv.readee (csvfile, delimitee=' ', quotechae='|').
for r in d:
        Print(', '.join(r)).
```

Result

The progeam has been executed and the output was verified.

Output

```
"['1',,'2',,'3']", "[33,,25,,56]", "[35,,30,,30]",
"[21,,40,,55]", "[71,,25,,55]", "[10,,10,,40]"
"['1',,'2',,'3']", "[33,,25,,56]", "[35,,30,,30]",
[21,,40,,55]", "[71,,25,,55]", "[10,,10,,40]".
```

Program No : 43.

Aim : Python programs to read specific coloumns. of a given csv file and print the content of the coloumns.

```
import csv,
with open('temp.csv', newline='') as csvfile:
data = csv.DictReader (csvfile).
Print ("ID Name")
 for row in data:
 Print ([row]['id'], row ['columns]'])
```

Result.

The program has been executed and the output was verified.

Output

```
ID Name

['1','2','3'] [33,25,56].
['1', 2','3'] [33,25,56].
['1','2','3'] [33,25,56].
['1','2','3'] [33,25,56].
```

Program No: 44

Aim: Python program to write a python dictionary to a csv file. After writing the csv file read the csv file and display the content

```
Import csv
csv columns= ['Id', 'Column1', Column 2', ' Column 3',
                    'Column 4', 'Column5']

dict-data = {'Id': ['1', '2', '3'].
  'Column1' : [33, 25, 56],
 'Column 2' : [35,30,30],
 'Column 3' : [21,40,55],
 'Column 4' : [71,25,55],
 'Column5' : [10,10,40]}.
 Csv-file = "temp.csv"
 try:
  with open (csv-file, 'w') as csvfile:
writer = csv. Dictwriter (csvfile, fieldnames = csv-columns)
writer. write header ()
for data in dict data:
writer. writerow (dict-data)
except  IOError:
   Print ("I/O error")
```

```
data = csv.DictReader (open(csv_file))
Print ("CSV file as a dictionary :\n")
for row in data:
    print (row.)
```

Result

The program has been executed and the output was verified.

Output

```
{'Id':"['1','2','3']", 'Column1':'[33,25,56]','Column2':
{'Id':"['1','2','3']", 'Column1':'[33,25,56]','Column2':,
{'Id':"['1','2','3']", 'Column1':'[33,25,56]','Column2':,
{'Id':"['1','2','3']", 'Column1':'[33,25,56]','Column2':
{'Id':"['1','2','3']", 'Column1':'[33,25,56]','Column2':,
```