

Output :-

Enter the value: 3

Area is 28.2600

Output:-

Enter the first number: 45

Enter the second number: 55

Enter the third number: 27

The largest number is 55

Date: 14-01-2021

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: "))  
print("Area is %6f" % area(num));
```

Result: The program has been executed and output was verified

Date: 16-01-2021

Program No: 2

Aim: Python program to find largest among 3 numbers

```
num1 = float(input("Enter the first number: "))  
num2 = float(input("Enter the second number: "))  
num3 = float(input("Enter the third number: "))  
  
if (num1 > num2) and (num1 > num3)  
    largest = num1  
elif (num2 > num1) and (num2 > num3)  
    largest = num2  
else  
    largest = num3
```

```
print("The largest number is ", largest)
```

Result: The program has been executed and output was verified.

Output :-

Enter an integer number : 5  
Square of 5 is 25

Output :-

The radius of the circle : 4  
The area of the circle with radius 4.0 is 50.2654

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

Date: 16-01-2021

Program No: 3

Aim: Python program to find square of a number

```
num = int(input("Enter an integer number: "))  
square = num * num  
print("Square of {num} is {square}")  
print
```

Result: The program has been executed and the output was verified.

Date: 26/01/2021

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 circle " + str(r) + " is: " + str(pi * r * r)).
```

Result: The program has been executed and the output was verified.

Date: 26-01-2021

Program No: 5

Aim: Python program to find square of n

```
list1 = [14, 20, 13, 8, 6, 2]  
for n in list1:  
    square = n * n  
    print(n, squared is, square)
```

Result: The program has been executed and the output was verified.

Output:-

Given string:

Hello.. How are you

The vowels present in the string

{ 'u', 'a', 'e', 'o' }

Output:-

{ 'python': 1, 'is': 1, 'a': 1, 'very': 1, 'versatile': 1, 'language': 1 }

Date: 26-01-2021

Program No: 6

Aim: Python program to find vowels in a string

```
stringA = "Hello.. How are you "
```

```
print ("Given string: \n", stringA)
```

```
vowels = "aAeEiIoOuu"
```

```
res = set([each for each in stringA if each in vowels])
```

```
print ('The vowels present in the string: \n', res)
```

Result: The program has been executed and the output was verified

Date: 26-01-2021

Program No: 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 in counts:
```

```
            counts[word] += 1
```

```
        else
```

```
            counts[word] = 1
```

```
    return counts
```

```
print(word_count('Python is a very versatile language'))
```

Result: The program executed and the output was verified.

Output:

Count of a in the list is: 5

Date: 26-01-2021

Program No: 8

Aim: Python program to count a in a list

```
a = ['anjali', 'reenu', 'gretta', 'aamy']
```

```
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.

Date: 26-01-2021

Program No: 9

Aim: Python program to check the length of list

```
list1 = [1, 2, 3, 4, 5, 6]
```

```
list2 = [9, 8, 7, 6, 3, 5]
```

```
len1 = len(list1)
```

```
len2 = len(list2)
```

```
if len1 == len2:
```

```
    print('both list have equal length')
```

```
else:
```

```
    print('both list doesn't have equal length')
```

Result: The program has executed and the output was verified.

Output:

both list have equal length

Output :

both list does not have equal sum

Output :

There are common elements

Date: 26-01-2021

Program No: 10

Aim: Python program to check the sum of list

```
list1 = [9, 4, 3, 7, 2]
```

```
list2 = [3, 4, 5, 8, 6]
```

```
total1 = sum(list1)
```

```
total2 = sum(list2)
```

```
if total1 == total2 :
```

```
    print('both list have equal sum')
```

```
else:
```

```
    print('both list does not have equal sum')
```

Result: The program has been executed and the output was verified

Date: 26-01-2021

Program No: 11

Aim: Python program to check the common elements in the list

```
list1 = [2, 3, 7, 5, 6, 9]
```

```
list2 = [8, 2, 3, 4, 5, 6]
```

```
for value in list1:
```

```
    if value in list2:
```

```
        common = 1
```

```
if common == 1:
```

```
    print('There are common elements')
```

```
else:
```

```
    print('No elements are common')
```

Result: The program has executed and the output was verified.

Output:

Oni\$n

Output:

nythop

Date: 27-01-2021

Program No: 12

Aim: Python program to replace a character

```
def change_char(str1):  
    char = str1[0]  
    str1 = str1.replace(char, '$')  
    str1 = char + str1[1:]  
    return str1  
print(change_char('onion'))
```

Result: The program has been executed and the output was verified

Date: 27-01-2021

Program No: 13

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

```
def change_string(str1):  
    return str1[-1:] + str1[1:-1] + str1[:1]  
print(change_string('python'))
```

Result: The program has been executed and the output was verified.

Output :-

None

{'d': 3, 'e': 2, 'l': 6, 'g': 5}

Output :

Original dictionary : {0:1, 2:3, 4:0, 3:4, 1:2}

Dictionary in ascending order by value : [(4,0), (0,1), (1,2), (2,3), (3,4)]

Dictionary in descending order by value : {5:4, 2:3, 1:2, 0:1, 4:0}

Date: 27-01-2021

Program No: 14

Aim: Python program to merge 2 dictionaries

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

dict1 = {'l': 6, 'g': 5}

dict2 = {'d': 3, 'e': 2}

print(Merge(dict1, dict2))

print(dict2)

Result: The program has been executed and the output was verified

Date: 27-01-2021

Program No: 15

Aim: Python program to ascent and descent dictionary

import operator

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

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)

Result: The program has been executed and the output was verified



Output :

[10, 13, 26, 29, 38, 50]

list after removing even numbers :

[13, 29]

Output :

GCD of 144 and 12 is 12

Date: 27-01-2021

Program No: 16

Aim: Python program to remove even numbers from the list

```
list = [10, 13, 26, 29, 38, 50]
```

```
print(list)
```

```
for i in list:
```

```
    if (i % 2 == 0):
```

```
        list.remove(i)
```

```
print("list after removing even numbers: ")
```

```
print(list)
```

Result: The program has been executed and the output was verified.

Date: 27-01-2021

Program No: 17

Aim: Python program to find gcd of number

```
def gcd(a, b):
```

```
    if (b == 0):
```

```
        return a
```

```
    return gcd(b, a % b)
```

```
a = 144
```

```
b = 12
```

```
if (gcd(a, b)):
```

```
    print('GCD of', a, 'and', b, 'is', gcd(a, b))
```

```
else:
```

```
    print('not found')
```

Result: The program has been executed and the output was verified



Output:

Enter a number: 5

The factorial of 5 is 120

Date: 03-02-2021

Program No: 18

Aim: Python program to find factorial of a number

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

```
factorial=1
```

```
if num<0:
```

```
    print("Sorry, factorial does not exist for negative number.")
```

```
elif num==0:
```

```
    print("The factorial of 0 is 1")
```

```
else:
```

```
    for i in range(1,num+1):
```

```
        factorial = factorial*i
```

```
    print("The factorial of ", num, "is", factorial)
```

Result: The program has been executed and the output was verified.

Output:

Enter the number of terms: 5

Fibonacci sequence:

0  
1  
1  
2  
3

Date: 03-02-2021

Programs No: 19

Aim: Python program to find fibonacci series of N terms

```
def recur_fibo(n):
```

```
    if n <= 1:
```

```
        return n
```

```
    else
```

```
        return (recur_fibo(n-1) + recur_fibo(n-2))
```

```
nterms = int(input("Enter the number of terms: "))
```

```
if nterms <= 0:
```

```
    print("Please enter a positive integer")
```

```
else:
```

```
    print("Fibonacci sequence: ")
```

```
    for i in range(nterms):
```

```
        print(recur_fibo(i))
```

Result: The program has been executed and the output was verified.

Output:

streaming

streamingly

Date: 03-02-2021

Program No: 20

Aim: Python program to perform string function

```
def add_string(stu1):  
    length = len(stu1)  
    if length > 1:  
        if stu1[-3:] == 'ing':  
            stu1 += 'ly'  
        else:  
            stu1 += 'ing'  
    return stu1  
print(add_string('stream'))  
print(add_string('streaming'))
```

Result: The program has been executed and the output was verified.

Output :

22

Output :-

Enter a number: 4444

Enter a number : 9999

4624

6084

6400

8464

Date : 03/02/2021

Program No: 21

Aim: Python program to perform the sum of given items

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

```
sum = sum(numbers)
```

```
print(sum)
```

Result: The program has been executed and the output was verified.

Date : 03/02/2021

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
```

```
                print(i)
```

Result: The program has been executed and the output was verified.

Output :-

Enter a number : 4

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

Date: 03/02/2021

Program No: 23

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

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

```
i = 1
```

```
j = 1
```

```
while i <= lines:
```

```
    j = 1
```

```
    while j <= i:
```

```
        temp = i * j
```

```
        print(temp, end = " ", flush = True)
```

```
        print(" ", end = " ", flush = True)
```

```
        j = j + 1
```

```
    print(" ")
```

```
    i = i + 1
```

Result: The program has been executed and the output was verified.

Output :-

```
{'h': 2, 'e': 2, 'l': 2, 'o': 3, ' ': 3, 'a': 1, 'y': 1, 'u': 1}
```

Date : 02/02/2021

Program No: 24

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

```
def char-frequency(str1):
```

```
    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 :-

Longest word : morning

length of the longest word : 7

Date : 03/02/2021

Program No: 25

Aim : Python program to accept a list of words and return length of longest words.

```
def find(word):
```

```
    w1 = []
```

```
    for n in word:
```

```
        w1.append(len(n), n))
```

```
    w1.sort()
```

```
    result = w1[-1][0], w1[-1][1]
```

```
    print("Longest word:", result[1])
```

```
    print("Length of the longest word:", result[0])
```

```
    find(['hello', 'morning', 'hi'])
```

Result: The program has been executed and the output was verified.



Output:-

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

Date: 03/02/2021

Program No: 26

Aim: Python program to constant Pattern using nested loop

def starc():

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(" ")

starc()

Result: The program has been executed and the output was verified.

Output :

The factors of 232 are

1  
2  
4  
8  
29  
58  
116  
232

Date : 03/02/2021

Program No: 27

Ans: Python program to print factors of a number

```
def print_factors(x):
```

```
    print("The factors of ", x, " are ")
```

```
    for i in range(1, x+1):
```

```
        if x%i==0:
```

```
            print(i)
```

```
print_factors(232)
```

Result: The program has been executed and the output was verified.

Output :-

Enter the length of a side of square

Enter the value : 2

Enter the length and breadth 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 is 4

Area of rectangle is 8

Area of triangle is 3.0

Date : 03/02/2021

Program No: 28

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

```
print('Enter the length of a 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 final year : 2040

Leap years:

2024

2028

2032

2036

2040

Date : 17/02/2021

Program No : 29

Aim : Python program to display future leap year 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 Leap Years:")
```

```
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:-

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

positive integer list: [1, 2, 9, 87, 24]

Date: 17/02/2021

Program No: 30

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

```
list1 = [1, -1, 2, -5, 9, -2, -54, 87, -33, -76, 24, -6]
```

```
pos = list()
```

```
for i in list1:
```

```
    if i > 0:
```

```
        pos.append(i)
```

```
print('Original list:', list1)
```

```
print('Positive integer list:', pos)
```

Result:- The program has been executed and the output was verified.