```
#  1. Program to demonstrate Using the Same Variable Name in Calling Function and Function De

def start(msg):
  print(msg)

msg="Hello"
start(msg)
```

```
     Hello
```

```
#  2. Program to demonstrate the Return of Multiple Values from a Function Definition#

def swap(x,y):
  return y,x

print(swap(7,8))
```

```
     (8, 7)
```

```
# 3.  Program to Check If a 3 Digit Number Is Armstrong Number or Not

from math import *

def armstrong(number):
  result = 0
  n = 0
  temp = number
  while (temp != 0):
    temp =int(temp / 10)
    n = n + 1
  temp = number
  while (temp != 0):
    remainder = temp % 10
    result = result + pow(remainder, n)
    temp = int(temp/10)
  if(result == number):
    print("Armstrong number")
  else:
    print("Not an Armstrong number")



number = int(input("Enter the number : "))
if(len(str(number))==3):
  armstrong(number)
else:
  print("Enter 3 digit number")
```

```
Enter the number : 371
Armstrong number
```

```
# 4.  Program to demonstrate the Scope of Variables

a="Global"

def fun(a):
  print(a)

print(a)
fun("Local")
```

```
Global
Local
```

```
# 5.  Calculate and Add the Surface Area of Two Cubes. Use Nested Functions

def SurfaceAreaOf2Cubes(a1,a2):
  A1=6*a1*a1
  print("Surface Area of First Cube is ",A1)

  def SAO2C(a2):
    A2=6*a2*a2
    print("Surface Area of Second Cube is ",A2)


    def Total(A1,A2):
      print("After adding ",A1+A2)

    Total(A1,A2)

  SAO2C(a2)


SurfaceAreaOf2Cubes(int(input("Enter Edge of First Cube :")),int(input("Enter Edge of First C
```

```
Enter Edge of First Cube :15
Enter Edge of First Cube :12
Surface Area of First Cube is  1350
Surface Area of Second Cube is  864
After adding  2214
```

```
# 6.  Program to demonstrate the Use of Keyword Arguments

def my_function(v1, v2, v3):
  print(v3)

my_function(v1=3, v2=5,v3=8)
my_function(v1=3, v2=5,v3=8)
```

```
8
```

```
# 7 is duplicate of #6
```

```
# 8.  Program to demonstrate the Use of *args and **kwargs

def adder(*num,**data):
    sum = 0

    for n in num:
        sum = sum + n
    print("Sum:",sum)

    for key, value in data.items():
        print("{} is {}".format(key,value))


adder(3,5,Firstname="K", Lastname="K", Age=222, Phone=1234567890)
```

```
Sum: 8
Firstname is K
Lastname is K
Age is 222
Phone is 1234567890
```

```
# 9.  Program to demonstrate Decorators in Python

def first(msg):
    print(msg)

def second(func, msg):
    func(msg)

second(first, "Hello!")
```

```
Hello!
```

```
# 10. Program to demonstrate Iterator in Python

arr={10,20,30,40,50,60,70}
sum=0
for i in arr:
  sum+=i
print(sum)
```

```
280
```

```
# 11. Program to demonstrate Generators in Python

def func(a):
    yield a
a=[1,2,3]

b=func(a)
next(b)
```

```
[1, 2, 3]
```

```
# 12.  Program to demonstrate lambda functions using filter() and map() function

tup= (5, 7, 22, 97, 54, 62, 77, 23, 73, 61)
print(tuple(map(lambda x: x+1 , tup)))

y = filter(lambda x: (x>=10), tup)
print(list(y))
```

```
(6, 8, 23, 98, 55, 63, 78, 24, 74, 62)
[22, 97, 54, 62, 77, 23, 73, 61]
```

```
# 13.  Program to demonstrate the use of math module and use of its various mathematical func

import math

print(math.pi)
print(math.e) #Euler's number
print(math.log(10))
print(math.exp(10))
print(math.e**10)
print(math.sqrt(9))
```

```
3.141592653589793
2.718281828459045
2.302585092994046
22026.465794806718
22026.465794806703
3.0
```

✓ 0s completed at 16:04 ● ✕