In [4]:
```
#Loop
listOfMovies=['Guntur Karam','Nela Ticket','Khushi','Athadu']

for movie in listOfMovies:
    lengthOfMovie=len(movie)
    print(lengthOfMovie)
```

```
12
11
6
6
****************************************
0
1
2
3
4
5
6
7
8
9
Good Bye!
```

In [7]:
```
print("****************************************")
count=0 #Outside the loop
while(count<10):
    #While loop block
    print(count)
    count=count+1 #increasing the value of count during each iterations
print("Good Bye!") #Outside the while loop
```

```
****************************************
0
1
2
3
4
5
6
7
8
9
Good Bye!
```

In [44]:
```
#while(True):
   # print("I am infinite loop")
```

In [10]:
```
print("******************list of movies using the while loop************************")
listOfMovies=['Guntur Karam','Nela Ticket','Khushi','Athadu']
noOfMovies=len(listOfMovies)
index=0
while(index<noOfMovies):
    print(listOfMovies[index])
    index=index+1

print("Thank you so much !!!")
```

```
******************list of movies using the while loop************************
Guntur Karam
Nela Ticket
Khushi
Athadu
Thank you so much !!!
```

In [15]:
```
fruits =['Banana','Mango','Guava','Oranges','Kiwi']
for fruit in fruits:
    print(fruit)
print("******************Using index of collection or sequence or above list in for loop******")
noOfFruits=len(fruits) #Number of items in fruits which is a list
print(range(noOfFruits))
for index in range(noOfFruits):
    print(index)
    fruit=fruits[index] #index value will change from 0 to 4
    print(fruit)
```

```
Banana
Mango
Guava
Oranges
Kiwi
******************Using index of collection or sequence or above list in for loop******
range(0, 5)
0
Banana
1
Mango
2
Guava
3
Oranges
4
Kiwi
```

In [23]:
```
#Nested Loop or loop inside a loop
count = 1
for i in range(10): #Outer loop
    #Inside outer loop body
    print (str(i) * i) # i=0 str(i) *i = 1
    for j in range(0, i): #Inner or Nested loop
        count = count +1
```

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

In [22]:
```
print(range(10))
for i in range(10):
    print(i)
    print(str(i) * i)
print ("Hi" * 1)
```

```
range(0, 10)
0

1
1
2
22
3
333
4
4444
5
55555
6
666666
7
7777777
8
88888888
9
999999999
Hi
```

In [24]:
```
print(str(0)*0)
print("****")
```

```
****
```

In [25]:
```
#List of list
list1=[1,2,3,4,5]
listOfList=[10,8,16,list1,10,15]
for item in listOfList:
    if(type(item) == type(list1)):
        #Display the items of list1
        for element in list1:
            print(element)
```

```
1
2
3
4
5
```

In [34]:
```
#Break -- using break we can come out of the loop
for i in range(5):
    if i == 3:
```

```
        break
    else:
        print("Lets take break of the all the classes")
    print(i)
print("Outside the loop")
listOfItems =[1,2,3,4,5,6,7,8,9,10]
for item in listOfItems:
    if item == 6:
      print("Eureka Eureka 6 is here")
      break
    print(item)
print("After Eureka Eureka !!!")
```

```
Lets take break of the all the classes
0
Lets take break of the all the classes
1
Lets take break of the all the classes
2
Outside the loop
1
2
3
4
5
Eureka Eureka 6 is here
After Eureka Eureka !!!
```

In [33]:

```
#Continue -- using continue we can skip the current execution of the loop
for i in range(5):
    if i == 3:
        continue
    else:
        print("I am enjoying continue")
    print(i)
print("Outside loop")
```

```
I am enjoying continue
0
I am enjoying continue
1
I am enjoying continue
2
I am enjoying continue
4
Outside loop
```

In [36]:
```
#Functions
#Defining the function named as displayMessage
def displayMessage():
    '''
        DocString is an optional
        displayMessage is a name of a function.
        This function doesn't take any arguments
        and it just prints the message in print() on standard output console
    '''
    print("Hello All Wish you happy sunday !!!")

#Calling the function
displayMessage()
print(displayMessage.__doc__)# __doc__ gives us the Doc String of the function
```

```
Hello All Wish you happy sunday !!!

        displayMessage is a name of a function.
        This function doesn't take any arguments
        and it just prints the message in print() on standard output console
```

In [38]:
```
def displayMsg(message):
    '''
      This displayMsg function takes one argument when it will be called

    '''
    print(message)

displayMsg("Hello All")
displayMsg() #This will give an error
```

```
Hello All
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[38], line 9
      6     print(message)
      8 displayMsg("Hello All")
----> 9 displayMsg()

TypeError: displayMsg() missing 1 required positional argument: 'message'
```

In [40]:
```
#addition function is defined
def addition(num1,num2):
    result = num1+num2
    return result

#addition function to be called
num1=15
num2=10
res=addition(num1,num2) #Calling the function and function returns the value of result
print(res)
```

```
25
```

In [41]:
```
def addition(num1,num2):
    return num1+num2
#addition function to be called
num1=15
num2=10
res=addition(num1,num2) #Calling the function and function returns the value of result
print(res)
```

```
25
```

In [43]:
```
#defining the function
def addition(num1,num2):
    '''
        This function doesn't return anything
    '''
    result=num1+num2
    #return result
#Use of addition function
num1=15
num2=10
res=addition(num1,num2) #Calling the function
print(res)
```

25

In [ ]: