In [ ]:
```
# Task 1:
```

1.1 : Write a Python Program to implement your own myreduce() function which works exactly likePython's built-in function reduce()

In [38]:
```
List1 = [1,2,3,4,5,6,7,8,9]

# Built-In Reduce fucntion

from functools import reduce
R1 = reduce(lambda a,b : a+b,List1)
R2 = reduce(lambda a,b : a*b,List1)

print(" R1 by using reduce() : ",R1)
print(" R2 by using reduce() : ",R2)
```

```
 R1 by using reduce() :  45
 R2 by using reduce() :  362880
```

In [39]:
```
# myreduce fucntion

def myreduce(myfunc,myseq):
    result=myseq[0]
    for i in myseq[1:]:
        result=myfunc(result,i)
    return result
```

In [40]:
```
R1 = myreduce(lambda a,b : a+b,List1)
R2 = myreduce(lambda a,b : a*b,List1)

print(" R1 by using myreduce() : ",R1)
print(" R2 by using myreduce() : ",R2)
```

```
 R1 by using myreduce() :  45
 R2 by using myreduce() :  362880
```

1.2 : Write a Python program to implement your own myfilter() function which works exactly like Python's built-in function filter()

In [41]:
```
List2 = [12,11,23,44,56,55,89,900,891,121,343]

# Built_in filter fucntion

even_list =list(filter(lambda a: a%2==0,List2))

print('even_list by using Built in filter() : ',even_list)
```

```
even_list by using Built in filter() :  [12, 44, 56, 900]
```

In [42]:
```
#myfiler() function

def myfilter(myfun,myseq):
    result = []
    for i in myseq:
        if myfun(i):
             result.append(i)
    return result
```

In [43]:
```
even_list =myfilter(lambda a: a%2==0,List2)

print('even_list by using Built in myfilter() : ',even_list)
```

```
even_list by using Built in myfilter() :  [12, 44, 56, 900]
```

2.Implement List comprehensions to produce the following lists.Write List comprehensions to produce the following Lists

In [44]:
```
# ['A', 'C', 'A', 'D', 'G', 'I', ’L’, ‘ D’]

L1=[ i for i in 'ACADGILD']
print(L1)
```

```
['A', 'C', 'A', 'D', 'G', 'I', 'L', 'D']
```

In [45]:
```
# ['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzz
z']

L2 = [i*j for i in 'xyz' for j in range(1,5)]
print(L2)
```

```
['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzz
z']
```

In [46]:
```
# ['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzz
z']

L3 = [ j*i for i in range(1,5) for j in 'xyz']
print(L3)
```

```
['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzz
z']
```

In [47]:
```
# [[2], [3], [4], [3], [4], [5], [4], [5], [6]]

L4 = [[i+j] for i in range(1,4) for j in range(1,4)]
print(L4)
```

```
[[2], [3], [4], [3], [4], [5], [4], [5], [6]]
```

In [48]:
```
# [[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]

L5 = [[i +j for i in range(1,5)] for j in range(1,5)]
print(L5)
```

```
[[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]
```

In [49]:
```
# [(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]

L6 = [(j,) for i in  range(1,4) for j in range(1,4)]
print(L6)
```

```
[(1,), (2,), (3,), (1,), (2,), (3,), (1,), (2,), (3,)]
```

3.Implement a function longestWord() that takes a list of words and returns the longest one.

In [53]:
```
def longestWord(myseq):
    maxlen = 0
    for i in myseq:
         if len(i) > maxlen:
                maxlen= len(i)
                maxword = i
    return maxword
```

In [54]:
```
iprlist = input('Please enter the list of the words : ')
iplist = iprlist.split(' ',iprlist.count(' '))

lonword = longestWord(iplist)
print('Longest word entered is {}'.format(lonword) )
```

```
Please enter the list of the words : This is my second python Assignment
Longest word entered is Assignment
```

In [ ]:
```
# Task 2:
```

1.1 : Write a Python Program(with class concepts) to find the area of the triangle using the below formula.area = (s*(s-a)*(s-b)*(s-c)) ** 0.5 Function to take the length of the sides of triangle from user should be defined in the parent class and function to calculate the area should be defined in subclass.

In [55]:
```
class Traingle():
    def getsides(self):
        self.a=float(input('Enter the first side of Traingle :'))
        self.b=float(input('Enter the second side of Traingle :'))
        self.c=float(input('Enter the Third side of Traingle :'))
    def __str__(self):
        return('Three side of traingle : {},{},{}'.format(self.a,self.b,self.c
))
```

In [56]:
```
class Area(Traingle):
    def cal_area(self):
        s= (self.a + self.b + self.c)/2
        area = (s*(s-self.a)*(s-self.b)*(s-self.c))**0.5
        print( 'Area of the Traingle is {:.3f} square unit.'.format(area))
        #  print(area)
```

In [57]:
```
A1= Area()
```

In [58]:
```
A1.getsides()
```

```
Enter the first side of Traingle :20
Enter the second side of Traingle :20
Enter the Third side of Traingle :10
```

In [59]:
```
print(A1)
A1.cal_area()
```

```
Three side of traingle : 20.0,20.0,10.0
Area of the Traingle is 96.825 square unit.
```

1.2 : Write a function filter_long_words() that takes a list of words and an integer n and returns the list of words that are longer than n.

In [60]:
```
def filter_long_words(myseq,n):
    result =[]
    for i in myseq:
        if len(i) > n :
            result.append(i)
    return result
```

In [61]:
```
wordlist = ['A','BB','CCC','DDDD','EEEEE','FFFFFFFFFFF','GGGGGG','KKKKKK']

n=3
flist= filter_long_words(wordlist,n)
print('Filter list with world greater than {} is {}'.format(n,flist))
```

```
Filter list with world greater than 3 is ['DDDD', 'EEEEE', 'FFFFFFFFFFF', 'GG
GGGG', 'KKKKKK']
```

2.1 : Write a Python program using function concept that maps list of words into a list of integers representing the lengths of the corresponding words .

In [62]:
```
def maplist(myseq):
    result = []
    for i in myseq:
        result.append(len(i))
    return result
```

In [63]:
```
L1 = ['ab','cde','erty']
no_L1 = maplist(L1)
print('Mapped List :{}'.format(no_L1))
```

```
Mapped List :[2, 3, 4]
```

2.2 : Write a Python function which takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.

In [64]:
```
def iden_vowel(cha):
    if cha.lower() in ('a','e','i','o','u'):
        return True
    else:
        return False
```

In [66]:
```
St1 = input('Enter the Character : ')

print(iden_vowel(St1[0]))
```

```
Enter the Character : F
False
```

In [67]:
```
St1 = input('Enter the Character : ')

print(iden_vowel(St1[0]))
```

```
Enter the Character : a
True
```

In [ ]:

In [ ]: