In [51]:

```
```

In [ ]:

```
                        ASSIGEMENT-2
```

In [ ]:

```
1.1)write a python program to implement your own myreduce( ) function which works excatly like pyth
on's built-in-function reduce().
```

In [ ]:

```
def myreduce(func,my_list):
    result=my_list[0]
    for item in my_list[2:]:
        result=func(result,item)
        return result
```

In [53]:

```
def sum(x,y):return x+y
print("sum on list [1,2,3,4,5,6] using reduce function" + str(myreduce(sum,[1,2,3,4,5,6])))
```

```
sum on list [1,2,3,4,5,6] using reduce function4
```

In [ ]:

```
1.2)write a python program to implement your own myfilter( ) function which works excatly like pyth
on's built-in-function filter().
```

In [39]:

```
def myfilter(func,my_list):
    result=[]
    for item in my_list:
        if func(item):
            result.append(item)

            return result
```

In [40]:

```
def ispositive(x):


    if(x<=0):
        return False
    else:
        return True


print("filter only positive integers on list[0,1,-2,3,4,5,7,7] using filter function"+str(myfilter
(ispositive,[0,1,-2,3,4,5,7,7])))
```

filter only positive integers on list[0,1,-2,3,4,5,7,7] using filter function[1]

In [ ]:

```
2)implement list comprehensions to produce the following lists.
write list comprrehensions to produce the following list
['A','C','A','D','G','I','L','D']
['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']
[['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzzz']]
[[2], [3], [4], [3], [4], [5], [4], [5], [6]]
[[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]
[(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]
```

In [194]:

```
word="ACADGILD"
a=[a for a in word]
print('ACADGILD===>>'+str(a))
```

```
ACADGILD===>>['A', 'C', 'A', 'D', 'G', 'I', 'L', 'D']
```

In [197]:

```
input_list=['x','y','z']
result=[item*num for item in input_list for num in range(1,5)]
print("['x','y','z']==>>"+str(result))
```

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

In [206]:

```
input_list=['x','y','z']
result=[[item*num for num in range(1,5) for item in input_list]]
print("['x','y','z']==>"+str(result))
```

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

In [209]:

```
input_list=[2,3,4]
result=[[item+num] for item in input_list for num in range(0,3)]
print("[2,3,4]==>"+str(result))
```

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

In [211]:

```
input_list=[2,3,4,5]
result=[[item+num for item in input_list] for num in range(0,4)]
print("[2,3,4,5]==>"+str(result))
```

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

In [99]:

```
input_list=[1,2,3]
result=[(b,a) for a in input_list for b in input_list]
print("[1,2,3]==>"+str(result))
```

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

In [ ]:

In [ ]:

```
3)Implement a function LongestWord() that takes a list of words and returns the Longest one.
```

In [37]:

```
def find_longest_word(words):
    return max(map(len,words))



if __name__=="__main__":
    print(find_longest_word(['small','biggest','a huge one here']))
```

```
15
```

In [38]:

```
a=[]
n=int(input("enter the number of words in list:"))
for x in range(0,n):
    word=input("enter word"+str(x+1)+":")
    a.append(word)
    max1=len(a[0])
    temp=a[0]
    for i in a:
        if(len(i)>max1):
            max1=len(i)
            temp=i

            print("the word with the longest one is:")

print(temp)
```

```
enter the number of words in list:4
enter word1:shaik
enter word2:roona
enter word3:anjum
enter word4:rasoolsahib
the word with the longest one is:
rasoolsahib
```

In [ ]:

```
TASK--2

1.1)write a python program (with class concepts) to find the area of the triangle using the below f
ormula.

area=(s*(s-a)*(s-b)*(s-c))**0.5

Function to take the lenght of a sides of triangle from the user should be defined in the parent cl
ass and function to calculate the area  should be defined in subclass.
```

In [36]:

```
a=float(input('enter the first side:'))
b=float(input('enter the second side:'))
c=float(input('enter the third side:'))
perimeter=a+b+c
s=(a+b+c)/2
area=(s*(s-a)*(s-b)*(s-c))**0.5
print('the perimeter of the triangle = %0.2f' %perimeter)
print('the semi perimeter of the triangle = %0.2f' %s)
print('the area of the triangle is %0.2f' %area)
```

```
enter the first side:3
enter the second side:6
enter the third side:8
```

```
the perimeter of the triangle = 17.00
the semi perimeter of the triangle = 8.50
the area of the triangle is 7.64
```

In [31]:

```
class polygon:
    def __init__(self,n):
        self.number_of_sides=n

    def print_num_sides(self):
        print("there are"+str(self.number_of_sides)+'sides.')
```

In [32]:

```
class Triangle(polygon):

    def __init__(self,lengths_of_sides):
        polygon.__init__(self,3)
        self.lengths_of_sides=lengths_of_sides

    def get_area(self):
        a,b,c=self.lengths_of_sides
        s=(a+b+c)/2
        return (s*(s-a)*(s-b)*(s-c))**0.5
```

In [33]:

```
p=polygon(9)
p.print_num_sides()
```

```
there are9sides.
```

In [34]:

```
tri=Triangle([3,4,5])
print(tri.get_area())
```

```
6.0
```

In [35]:

```
tri.print_num_sides()
```

```
there are3sides.
```

In [1]:

```
1.2)write a function filter_long_words() that takes a list of words and an integer  n and returns t
he list of words that are longer than n.
```

In [29]:

```
def filterlongword(word,length):
    return (word for word in word if len(word)>=length)
```

In [30]:

```
def main():
    word=input("please enter the words,separated by space:").split()
    length=int(input("minimum length of words to keep:"))
    print("words longer than {} are {}.".format(length,','.join(filterlongword(word,length))))
```

```
main()
```

```
please enter the words,separated by space:a bb ccc dddd eeeee ffffff hhhhhhh
minimum length of words to keep:3
words longer than 3 are ccc,dddd,eeeee,ffffff,hhhhhhh.
```

In [10]:

```
2.1) write apython porgram using function concept that maps list of words into a list of integers r
epresenting the lenghts of the corresponding words.

hint: If a list ['ab','cde','erty'] is passed on to  the python function output should come as [2,3
,4]
    Here,2,3 and 4 are the lengths of the words in the list.
```

In [26]:

```
def map_to_lengths_for(words):
    lengths=[]
    for word in words:
        lengths.append(len(word))
        return lengths
```

In [27]:

```
def map_to_lengths_map(words):
    return map(len,words)
```

In [28]:

```
def map_to_lengths_lists(words):
    return[len(word) for word in words]


if __name__=="__main__":
    words=['ab','cde','erty']
    print(map_to_lengths_for(words))
    #print(map_to_lengths_map(words))
    print(map_to_lengths_lists(words))
```

```
[2]
[2, 3, 4]
```

In [ ]:

```
2.2)Write a python function which takes a character(i.e. a string of length 1) and return True if i
t is a vowel,False otherwise.
```

In [25]:

```
def is_vowel(char):
    vowels=('a','e','i','o','u')
    if char not in vowels:
        return False
    return True
if __name__=="__main__":
    print(is_vowel(1))
    print(is_vowel('a'))
    print(is_vowel('b'))
    print(is_vowel('o'))
    print(is_vowel('z'))
```

```
False
True
False
True
False
```

In [ ]:

```

```

In [ ]:

```

```

In [ ]:

```

```

In [ ]:

```

```