1.1 TASK 1

Write a Python Program to implement your own myreduce() function which works exactly like Pyth

```
from functools import reduce
#Multiply all numbers in a list

data = [2,3,4,5,6,7,8,9,10,11,12,14,15]

mult = lambda x,y: x*y

reduce(mult, data)
```

```
100590336000
```

```
from functools import reduce
#Multiply all numbers in a list

data = [2,3,4,5,6,7,8,9,10,11,12,14,15]

reduce(lambda x,y: x*y,data)
```

```
100590336000
```

1.2 Write a Python program to implement your own myfilter() function which works exactly like Pyt

```
def myfilter(num):
    if num%2 != 0:
        return True

numbers = [11,12,13,14,15,16,17,18,19,20,21,22,23,35,32,]
list (filter(myfilter, numbers))
```

```
[11, 13, 15, 17, 19, 21, 23, 35]
```

2. Implement List comprehensions to produce the following lists. Write List comprehensions to pr
’L’, ‘ D’] ['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz'] ['x', 'y', 'z', 'xx', 'yy', 'zz', 'xx', 'yy', 'zz', 'x
[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)]

```
a = ['A', 'C', 'A', 'D', 'G', 'I', "’L’", "‘ D’"]
b = ['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']
c = ['x', 'y', 'z', 'xx', 'yy', 'zz', 'xx', 'yy', 'zz', 'xxxx', 'yyyy', 'zzzz']
```

```
d = [[2], [3], [4], [3], [4], [5], [4], [5], [6]]
e = [[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]
f = [(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]

collections = [a,b,c,d,e,f]
me = [i for i in collections]
print(me)
```

```
:', 'A', 'D', 'G', 'I', '’L’', '‘ D’'], ['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'y
```

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

```
def longlen(*strings):

        return max(strings, key=len)

longlen('kikkkaaaa','sule', 'scDWD', 'mmm','dhgsohwitirr')
```

```
'dhgsohwitirr'
```

TASK 2 1.1

Write a Python Program(with class concepts) to find the area of the triangle using the below formu

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 defined in subclass.

```
class triangle():
    def __init__(self,a,b,c):
        self.a = a
        self.b = b
        self.c = c
    def area(self):
        self.s=(self.a + self.b + self.c)/2
        return (self.s*(self.s-self.a)*(self.s-b)*(self.s-self.c)) ** 0.5
print("To find the Area of a triangle: ")
a = int(input("Please enter the length of the  1st side of triangle: "))
b = int(input("Please enter the length of the 2nd side of triangle: "))
c = int(input("Please enter the length of the 3rd side of triangle: "))
r1 = triangle(a,b,c)

print ("Area: ",r1.area())
```

```
To find the Area of a triangle:
Please enter the length of the  1st side of triangle: 43
Please enter the length of the 2nd side of triangle: 22
Please enter the length of the 3rd side of triangle: 34
Area:  370.3321988431468
```

1.2

Write a function filter_long_words() that takes a list of words and an integer n and returns the list c

```
def filterlongword(number,*string):
    return [word for word in string if len(word) > number]
filterlongword(3,'myna', 'asdadasdas','fhwroighretgije0ht')
```

```
['myna', 'asdadasdas', 'fhwroighretgije0ht']
```

2.1

Write a Python program using function concept that maps list of words into a list of integers repre words. Hint: If a list [ab,cde,erty] is passed on to the python function output should come as [2,3,4 in the list

```
def my_word(r):
    return len(r)
words = ['python', 'java', 'scala', 'hadoop']

list(map(my_word, words))
```

```
[6, 4, 5, 6]
```

2.2 Write a Python function which takes a character (i.e. a string of length 1) and returns True if it i

```
def status(alpha):
    vowels = ('a', 'e', 'i', 'o', 'u')
    if alpha not in vowels:
        return False
    return True
status('a')
```

```
True
```