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

like Python's built-in function reduce()

In [6]:
```
def myreduce(fun,alist):
    x=alist[0]
    for i in range(1,len(alist)):
        x=fun(x,alist[i])
    return x

def summ(x1,x2):
    return x1+x2

def product(x1,x2):
    return (x1*x2)

a=[1,2,3,4,5]

print("Sum of all elements in list: ",myreduce(summ,a))
print("Product of all elements in list: ",myreduce(product,a))
```

```
Sum of all elements in list:  15
Product of all elements in list:  120
```

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

like Python's built-in function filter()

In [16]:
```
def myfilter(fun,values):
    result=list()
    for i in values:
        if fun(i):
            result.append(i)
    return result
```

In [17]:
```
def fun(x):
    if x%5==0:
        return True
    else:
        return False

def odd_numbers(x):
    if x%2==1:
        return True
    else:
        return False


a=[1,2,5,10,15]

print(myfilter(fun,a))
print(myfilter(odd_numbers,a))
```

```
[5, 10, 15]
[1, 5, 15]
```

2. Implement List comprehensions to produce the following lists.

Write List comprehensions to produce the following Lists

1)['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']

In [5]:
```
list1=[i*j for j in ["x","y","z"] for i in range(1,5)]
list1
```

Out[5]:
```
['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']
```

2)['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzzz']

In [6]:
```
list2=[i*j for j in range(1,5) for i in ["x","y","z"]]
list2
```

Out[6]:
```
['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzzz']
```

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

In [19]:
```
list3=[[i+j] for j in [0,1,2] for i in range(2,5)]
list3
```

Out[19]:
```
[[2], [3], [4], [3], [4], [5], [4], [5], [6]]
```

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

In [20]:
```
list4=[[i+j for j in [0,1,2,3]] for i in range(2,6)]
list4
```

Out[20]:
```
[[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]
```

5) [(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]

In [21]:
```
list5=[(i,j) for j in range (1,4) for i in range(1,4)]
list5
```

Out[21]:
```
[(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]
```

In [ ]:
```

```