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

In [1]:
```
def myreduce(func, arg1):
    x=arg1[0]
    for i in arg1[1:]:
        x = func(x,i)
    return x
print(myreduce(lambda a,b: a+b,[1,2,3,4,5,6]))
```

```
21
```

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

In [2]:
```
def myfilter(func, arg1):
    x=[]
    for i in arg1:
        if func(i):
            x=x+[i]
    return x
myfilter(lambda item:len(item)%2==0,["Manoj","Kumar","Mishra","is","a","cool","gu
```

Out[2]:
```
['Mishra', 'is', 'cool']
```

## 2. Implement List comprehensions to produce the following lists.

Write List comprehensions to produce the following Lists

['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', 'xx', 'yy', 'zz', '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 [3]:
```
print([x.upper() for x in 'acadgild'])
print([x*y for y in 'xyz' for x in range(1,5)])
print([x*y for y in range(1,5) for x in 'xyz'])
print([[x+y] for y in range(1,4) for x in range(1,4)])
print([[x+y for y in range(1,5)] for x in range(1,5)])
print([(x,y) for y in range(1,4) for x in range(1,4)])
```

```
['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)]
```

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

In [4]:
```
def longestWord(lis):
    return myreduce(lambda a,b: b if len(a)<len(b) else a,lis)
longestWord(["Manoj","Kumar","Mishra","is","a","cool","guy"])
```

Out[4]: 'Mishra'

In [ ]:
```

```