In [42]:

```
#Function Practice

#write a python function that takes a list of numbers as input and returns the sum of all even numbers in

list1 = []

number = int(input("How many elements you want to add: "))
def evenSum(list1):
    Sum = 0
    for i in range(0,number):
        a = int(input())
        list1.append(a)
        #print(list1)

    for x in list1:
        if x % 2 == 0:
            Sum += x
    print("Sum of the numbers is: ",Sum)


evenSum(list1)
```

```
How many elements you want to add: 5
1
2
5
6
8
Sum of the numbers is:  16
```

In [140]:
```
#Create a function that takes a string and return vowels(a,e,i,o,u)

str = input("Enter the string: ")
def vowelFunc(str):
    vowels = []
    for x in str:
        if x == 'a' or x == 'e' or x == 'i' or x =='o' or x == 'u':
            vowels.append(x)
    print(vowels)

vowelFunc(str)
```

```
Enter the string: india
['i', 'i', 'a']
```

In [85]:
```
#Write a function that finds the maximum element in a list of numbers without using the max() function

list1 = []
count = int(input("How many elements are there in a list: "))

def maxNum(list1):

    for i in range(0,count):
        a = int(input())
        list1.append(a)
        #print(list1)


    list1.sort()
    print("Largest number is: ",list1[-1])

maxNum(list1)
```

```
How many elements are there in a list: 5
9
3
67
34
2
Largest number is:  67
```

Generator Practice

In [95]:
```python
#Create a generator function that yeilds Fibonacci numbers one at a time

def fibonacci_sequence():
    num1,num2 = 1,1
    while True:
        yield num1
        num1,num2 = num2, num1+num2

g = fibonacci_sequence()
for i in range(10):
    print(g.__next__())
```

```
1
1
2
3
5
8
13
21
34
55
```

In [112]:

```
#Write a generator that generates sequence of prime numbers

def primeNum(start,end):
    while True:
        for i in range(start,end):
            if(i % 2 == 0):
                break
            else:
                yield i

start = int(input("Enter the starting point: "))
end = int(input("Enter the ending point: "))
prime_Generator = primeNum(start,end)

for i in range(0,end):
    print(next(prime_Generator))
```

```
Enter the starting point: 6
Enter the ending point: 20

---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
Cell In[112], line 16
     13 prime_Generator = primeNum(start,end)
     15 for i in range(0,end):
---> 16     print(next(prime_Generator))

Cell In[112], line 4, in primeNum(start, end)
      3 def primeNum(start,end):
----> 4     while True:
      5         for i in range(start,end):
      6             if(i % 2 == 0):

KeyboardInterrupt:
```

In [108]:
```python
#Implement a generator that generates random numbers within a specified range

import random

def randomNum(start,end):
    while True:
        yield random.randint(start,end)

start = int(input("Enter the starting point: "))
end = int(input("Enter the ending point: "))

random_Generator_Obj = randomNum(start,end)

for i in range(0,end):
    print(next(random_Generator_Obj))
```

```
Enter the starting point: 5
Enter the ending point: 20
5
20
12
20
12
5
8
17
5
20
18
7
12
15
10
13
13
7
14
7
```

Map and filter practice

In [117]:
```
#use map to convert a list of strings to uppercase

x = list(map(lambda a : a.upper(),["maths",'physics','science','chemistry'] ))
print(x)
```

```
['MATHS', 'PHYSICS', 'SCIENCE', 'CHEMISTRY']
```

In [118]:
```
#Use filter to find all prime numbers ina list of numbers

def primeNum(num):
    if num < 2:
        return False
    for i in range(2, int(num ** 0.5) + 1):
        if num % i == 0:
            return False
    return True

numbers = [23,78,89,3,7,5,86]
prime_numbers = list(filter(primeNum, numbers))
print(prime_numbers)
```

```
[23, 89, 3, 7, 5]
```

In [122]:
```
#Use map and filter together to calculate the square of all even numbers in list

def evenNum(num):
    if num % 2 == 0:
        return True
    else:
        return False

list1 = [2,6,3,4,9,88]

even_numbers = list(filter(evenNum,list1))
print(even_numbers)

square_of_numbers = list(map(lambda x : x*x,even_numbers))
print("Square of numbers: ",square_of_numbers)
```

```
[2, 6, 4, 88]
Square of numbers:  [4, 36, 16, 7744]
```

Reduce Practice

In [124]:
```
#Use reduce to find the product of all numbers in a list

from functools import reduce

x = list(range(1,10))

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

Out[124]: 362880

In [139]:
```
#Write a program that uses reduce to fin the factorial of a given number
from functools import reduce

def factNum(num):
    if num == 0:
        return False
    else:
        return reduce(lambda x,y : x*y,range(1,num+1))

factNum(4)
```

Out[139]: 24

In [ ]:
```

```

In [ ]:
```

```