REGex Software Services

Data Science Intern

Task 2

Q1. Write a lambda expression to extract first word of a string.

In [2]:
```
inpt = input("Type a sentence: ")
l = lambda str1:str1.split(" ")[0]
print(l(inpt))
```

```
Type a sentence: kiransf kfjd f
kiransf
```

Q2. Write a function to extract first word of s string (with many words separated by space).

In [4]:
```
def first_word(l):
  return l.split(" ")[0]

inpt2 = input("Type a sentence: ")
first_word(inpt2)
```

```
Type a sentence: Joker is always unique
```

Out[4]:
```
'Joker'
```

Q3. Extract the first word from every string from a list of strings by using map function.

In [9]:
```
trio = ['John Wick 1', 'John Wick 2', 'John Wick 3']
list(map(lambda s: s.split()[0], trio))
```

Out[9]:
```
['John', 'John', 'John']
```

Q4. Write a function to return a list of prime factors of a given number.

In [13]:
```
import math

num = int(input('Enter a number: '))

def prime_factors(n):
    ln =[ev for ev in range(2,n+1) if n%ev == 0]
    def prime(x):
        for i in range(2,int(math.sqrt(x))+1):
            if x%i==0:
                return False
        else:
            return True

    print('list of prime factors: ',list(filter(prime,ln)))
prime_factors(num)
```

```
Enter a number: 34
list of prime factors:  [2, 17]
```

Q5. Write a function that finds 2nd largest among 4 numbers (Repetitions are allowed, without sorting).

In [17]:
```
def second_largest(n):
    max_num = max(n[0],n[1])
    second_larg_num = min(n[0],n[1])

    for i in range(2,len(n)):
        if n[i] > max_num:
            second_larg_num = max_num
            max_num = n[i]
        elif n[i] > second_larg_num and n[i]!= max_num:
            second_larg_num = n[i]
    return second_larg_num

n=[45,34,34546,12,3454,1,23563,1]
result= second_largest(n)
print(result)
```

```
23563
```