RENAME KEY CITY TO LOCATION IN THE FOLLOWING DICTIONARY:

In [12]:
```
skillSanta_Dict = {
  "name": "sachin",
  "age":22,
  "salary": 60000,
  "city": "New delhi"}

keys = ["name", "city"]

newDict = {k: sampleDict[k] for k in keys}
print(newDict)
```

```
{'name': 'sachin', 'city': 'New delhi'}
```

OCCURANCE OF EACH ELEMENT

In [15]:
```
sampleList = [11, 45, 8, 11, 23, 45, 23, 45, 89]
print("Original list ", sampleList)

countDict = dict()
for item in sampleList:
  if(item in countDict):
    countDict[item] += 1
  else:
    countDict[item] = 1

print("Printing count of each item  ",countDict)
```

```
Original list  [11, 45, 8, 11, 23, 45, 23, 45, 89]
Printing count of each item   {11: 2, 45: 3, 8: 1, 23: 2, 89: 1}
```

REMOVE DUPLICATE FROM A LIST AND CREATE A TUPILE AND FIND THE MINIMUM AND MAXIMUM NUMBER:

In [16]:
```
sampleList = [87, 52, 44, 53, 54, 87, 52, 53]

print("Original list", sampleList)

sampleList = list(set(sampleList))
print("unique list", sampleList)

tuple = tuple(sampleList)
print("tuple ", tuple)

print("Minimum number is: ", min(tuple))
print("Maximum number is: ", max(tuple))
```

```
Original list [87, 52, 44, 53, 54, 87, 52, 53]
unique list [44, 52, 53, 54, 87]
tuple  (44, 52, 53, 54, 87)
Minimum number is:  44
Maximum number is:  87
```

CREATE A FUNCTION SHOWEMPLOYEE()

In [17]:
```
def showEmployee(name, salary=50000):
    print("Employee", name, "salary is:", salary)

showEmployee("eddy", 50000)
showEmployee("eddy")
```

```
Employee eddy salary is: 50000
Employee eddy salary is: 50000
```

INNER FUNCTION TO CALCULATE THE ADDITION:

In [18]:
```
def outerFun(a, b):
    square = a**2
    def innerFun(a,b):
        return a+b
    add = innerFun(a, b)
    return add+5

result = outerFun(5, 10)
print(result)
```

```
20
```

RECURSIVE FUNCTION TO PRINT THE FIBONACCI SERIES OF N NUMBERS:

In [19]:
```
def recur_fibo(n):
   if n <= 1:
       return n
   else:
       return(recur_fibo(n-1) + recur_fibo(n-2))

nterms = 10

if nterms <= 0:
   print("Plese enter a positive integer")
else:
   print("Fibonacci sequence:")
   for i in range(nterms):
       print(recur_fibo(i))
```

```
Fibonacci sequence:
0
```

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

ASSIGN A DIFFERENT NAME TO FUNCTION:

In [20]:
```
def displayStudent(name, age):
    print(name, age)

displayStudent("shiva", 22)

showStudent = displayStudent
showStudent("shiva", 22)
```

```
shiva 22
shiva 22
```

GET PROPER NUMBER STOP ASKING

In [24]:
```
number1 = input("Enter number ")
number2 = input("Enter another number ")

print("\n")
print("Printing type of input value")
print("type of number ", type(number1))
print("type of number_two ", type(number2))
```

```
Enter number 7550177083
```

```
Enter another number shiva


Printing type of input value
type of number  <class 'str'>
type of number_two  <class 'str'>
```

PYTHON FUNCTION THAT ACCEPTS A STRING AND CALCULATE THE NUMBER OF UPPER AND LOWER CASE LETTERS:

In [27]:
```
def string_test(s):
    d={"UPPER_CASE":0, "LOWER_CASE":0}
    for c in s:
        if c.isupper():
           d["UPPER_CASE"]+=1
        elif c.islower():
           d["LOWER_CASE"]+=1
        else:
           pass
    print ("Original String : ", s)
    print ("No. of Upper case characters : ", d["UPPER_CASE"])
    print ("No. of Lower case Characters : ", d["LOWER_CASE"])

string_test('The quick Brown Fox')
```

```
Original String :  The quick Brown Fox
No. of Upper case characters :  3
No. of Lower case Characters :  13
```

PYTHON FUNCTION TO CHECK WHETHER A NUMBER IS PERFECT OR NOT:

In [28]:
```
def perfect_number(n):
```

```
    sum = 0
    for x in range(1, n):
        if n % x == 0:
            sum += x
    return sum == n
print(perfect_number(6))
```

```
True
```