In [1]:
```
#dict.clear()
#Removes all elements of dictionary dict
dict = {'Name': 'Zara', 'Age': 7};
print("Start Len : %d" % len(dict))
dict.clear()
print("End Len : %d" % len(dict))
```

```
Start Len : 2
End Len : 0
```

In [2]:
```
#dict.items()
#Returns a list of dict's (key, value) tuple pairs
dict = {'Name': 'Zara', 'Age': 7}
print ("Value : %s" % dict.items())
```

```
Value : dict_items([('Name', 'Zara'), ('Age', 7)])
```

In [3]:
```
# dict.copy()
#Returns a copy of dictionary dict
dict1 = {'Name': 'Zara', 'Age': 7};
dict2 = dict1.copy()
print ("New Dictinary : %s" % str(dict2))
```

```
New Dictinary : {'Name': 'Zara', 'Age': 7}
```

In [5]:
```
#Python Program to Generate a Dictionary that Contains Numbers (between 1 a
n=int(input("Enter a number:"))
d={x:x*x for x in range(1,n+1)}
print(d)
```

```
Enter a number:5
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
```

In [6]:
```
#Python Program to Sum All the Items in a Dictionary
d={'A':100,'B':540,'C':239}
print("Total sum of values in the dictionary:")
print(sum(d.values()))
```

```
Total sum of values in the dictionary:
879
```

In [1]:
```
#Python Program to Remove the Given Key from a Dictionary
d = {'a':1,'b':2,'c':3,'d':4}
print("Initial dictionary")
print(d)
key=input("Enter the key to delete(a-d):")
if key in d:
    del d[key]
else:
    print("Key not found!")
    exit(0)
print("Updated dictionary")
print(d)
```

```
Initial dictionary
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
Enter the key to delete(a-d):d
Updated dictionary
{'a': 1, 'b': 2, 'c': 3}
```

In [4]:
```
# Keys and Values example
d={}
print (" The dictionary elements are")
for i in range(1,21):
    d[i]=i**2
print (d)
# To print key and values
print (" Key==> Value are")
```

```
for (k,v) in d.items():
    print(k,"==>",v)
# To print key only
print ("\nTo print key only")
for k in d.keys():
    print(k, end=" ")
#To print value only
print ("\nTo print values only")
for v in d.values():
    print(v, end=" ")
```

```
 The dictionary elements are
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121,
12: 144, 13: 169, 14: 196, 15: 225, 16: 256, 17: 289, 18: 324, 19: 361, 20: 40
0}
 Key==> Value are
1 ==> 1
2 ==> 4
3 ==> 9
4 ==> 16
5 ==> 25
6 ==> 36
7 ==> 49
8 ==> 64
9 ==> 81
10 ==> 100
11 ==> 121
12 ==> 144
13 ==> 169
14 ==> 196
15 ==> 225
16 ==> 256
17 ==> 289
18 ==> 324
19 ==> 361
20 ==> 400

To print key only
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
To print values only
1 4 9 16 25 36 49 64 81 100 121 144 169 196 225 256 289 324 361 400
```

In [2]:

```
#Python Program to add name and mark as key->value pair in a Dictionary and
n=int(input("Enter no of records"))
d={}
for i in range(1,n+1):
    name= input("Enter name %d"%(i))
    mark=int(input("Enter mark %d"%(i)))
    d[name]=mark
    print (d)
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
/opt/anaconda3/lib/python3.7/site-packages/ipykernel/kernelbase.py in _input_r
equest(self, prompt, ident, parent, password)
    884             try:
--> 885                 ident, reply = self.session.recv(self.stdin_socket, 0)
    886             except Exception:

/opt/anaconda3/lib/python3.7/site-packages/jupyter_client/session.py in recv(s
elf, socket, mode, content, copy)
    802         try:
--> 803             msg_list = socket.recv_multipart(mode, copy=copy)
    804         except zmq.ZMQError as e:

/opt/anaconda3/lib/python3.7/site-packages/zmq/sugar/socket.py in recv_multipa
rt(self, flags, copy, track)
    474         """
--> 475         parts = [self.recv(flags, copy=copy, track=track)]
    476         # have first part already, only loop while more to receive

zmq/backend/cython/socket.pyx in zmq.backend.cython.socket.Socket.recv()

zmq/backend/cython/socket.pyx in zmq.backend.cython.socket.Socket.recv()

zmq/backend/cython/socket.pyx in zmq.backend.cython.socket._recv_copy()

/opt/anaconda3/lib/python3.7/site-packages/zmq/backend/cython/checkrc.pxd in z
mq.backend.cython.checkrc._check_rc()

KeyboardInterrupt:

During handling of the above exception, another exception occurred:

KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-2-52034f353dba> in <module>
      1 #Python Program to add name and mark as key->value pair in a Dictionar
y and print it.
----> 2 n=int(input("Enter no of records"))
      3 d={}
      4 for i in range(1,n+1):
      5     name= input("Enter name %d"%(i))
```

In [1]:

```
# Keys and Values example
d={}
print (" The dictionary elements are")
for i in range(1,21):
    d[i]=i**2
print (d)
# To print key and values
print (" Key==> Value are")
for (k,v) in d.items():
    print(k,"==>",v)
# To print key only
print ("\nTo print key only")
for k in d.keys():
    print(k, end=" ")
#To print value only
print ("\nTo print values only")
for v in d.values():
    print(v, end=" ")
```

```
 The dictionary elements are
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121,
12: 144, 13: 169, 14: 196, 15: 225, 16: 256, 17: 289, 18: 324, 19: 361, 20: 40
0}
 Key==> Value are
1 ==> 1
2 ==> 4
3 ==> 9
4 ==> 16
5 ==> 25
6 ==> 36
7 ==> 49
8 ==> 64
9 ==> 81
10 ==> 100
11 ==> 121
12 ==> 144
13 ==> 169
14 ==> 196
15 ==> 225
16 ==> 256
17 ==> 289
```

In [1]:

```
#Python program to convert 2 digit number into words
d={0:'',1:'one',2:'two',3:'three',4:'four',5:'five',6:'six',7:'seven',8:'ei
    10:'ten',11:'eleven',12:'twelve',13:'thirteen',14:'fourteen',15:'fiftee
    17:'seventeen',18:'eightteen',19:'nineteen',20:'twenty',30:'thirty',40:
    60:'sixty',70:'seventy',80:'eighty',90:'ninty'}
num=int(input("Enter the integer between 1 to 99:"))
if (num<=20):
    print(d[num])
if(num>20 and num<100):
    if num%10==0:
        print(d[num])
    else:
        print(d[num//10*10]+" "+d[num%10])
```

```
Enter the integer between 1 to 99:25
twenty five
```

In [2]:

```
#How to sort a dictionary by values in Python
d = {"Pierre": 42, "Anne": 33, "Zoe": 24}

#Use the sorted function and operator module
import operator
sorted_d = sorted(d.items(), key=operator.itemgetter(1))
print(sorted_d)
sorted_a= sorted(d.items(), key=operator.itemgetter(1),reverse=True)
print(sorted_a)
```

```
[('Zoe', 24), ('Anne', 33), ('Pierre', 42)]
[('Pierre', 42), ('Anne', 33), ('Zoe', 24)]
```

In [ ]:

```

```