Sorting

- It is the process of making objects in the collections either in ascending or decending

List Collection:

- L.sort(key=None, reverse=False)
  - Key represents based which key element do you want sort the object
  - Value for the key should be lambda expression
  - Reverse attribute represents Ascending or decending
  - Default value for reverse is “False” it represent “Ascending order”

Example:

```
#sorting
#L.sort(key=None,reverse=False)

lst=[30,20,10,40,50]
print("List : ",lst)

lst.sort()
print("After sorting : ",lst)

lst.sort(reverse=True)
print("Result is : ",lst)
```

Example 2:

```
#sorting
#L.sort(key=None,reverse=False)  valid
#T.sort( ) AttributeError
#S.sort( ) AttributeError
#D.sort( ) AttributeError
#S.sort( ) AttributeError  here s is String obj


'''
t=(10,20,40,30,50)
t.sort()  #AttributeError '''

'''
s={30,20,10,40,50}
s.sort() #AttributeError '''
```

#sorted(iterable,key=None,reverse=False) -> list

- Sorted() is predefined function from built-in Module ,it is used for making the objects of iterable [str | list | set | tuple | dict] collection in sorting order

Example:

```
#sorted(iterable,key=None,reverse=False) -> list
lst=[20,30,10,2,5,50]
print("List : ",lst)
print("Sorting the list")
slst=sorted(lst,key=None,reverse=False)
print("Result is : ",slst)
print("- "*30)

t=(40,30,50,20,10)
print("Tuple : ",t)
t=tuple( sorted(t) )   #tuple(iterable) -> tuple
print("Result : ",t)
t=tuple(sorted(t,reverse=True))
print("DEC : ",t)
```

Example :

```
s={40,30,20,10,50}
print("Set Collection ",s)
print("Sorting the Set Collection ")
lst=sorted(s)  #set(iterable) -> set
print("After sorting ",lst)
```

Example :

```
s="AEDCBF"
print("Actual String : ",s)
```

```
print("String Sorting")
lst=sorted(s)

#S.join(iterable) -> str For converting list to string
s="".join(lst)
print("Result is : ",s)


Example:
#sorted(iterable,key=None,reverse=False) -> list
#D.sort() AttributError

d={'a':'apple','c':'cat','d':'doll','b':'ball'}
print("Dict : ",d)

lst=sorted(d) # it will take all keys and sorting
print("Sorted keys : ",lst)

lst_v=sorted(d.values())
print("Sorted values : ",lst_v)

lst_items=sorted(d.items())
print("Sorted items : ",lst_items)
```

```
#sorted(iterable,key=None,reverse=False) -> list

import time
lst=[("nani",30),("anu",25),("cnu",32),("priya",26)]

print("Before sorting ")
for i in lst:
    time.sleep(1)
    print(i)

print("sorting based on names ")
lst_names=sorted(lst,key=lambda item:item[0])
for i in lst_names:
    time.sleep(1)
    print(i)

print("soring based on age ")
lst_age=sorted(lst,key=lambda item : item[1])
for i in lst_age:
    time.sleep(1)
    print(i)
```