13/1/2022

List Data type(list()):-

```
>>> l  = [12,89,'gh',78.89,[56,89,90],(56,89),89]
>>> type(l)
<class 'list'>
>>> l[0]
12
>>> l[-1]
89
>>> l[1]
89
>>> l[-3]
[56, 89, 90]
>>>
```

If we want to represent a group of values as a single entity where insertion order required to preserve and duplicate are allowed then we should go for list data type.Here every data are separet by comma.List is enclosed with in squre bracket[].

1. Insertion order is preserved.
2. Heterogeneous object are allowed.
3. duplicate are allowed.
4. Growable in nature(add data).
5. Values should be enclosed with in squre bracket.

Note:- List is dynamic because based on our requirement we can increase the size and decrease the size.

Important Function of list:-

```
>>> l = [10,20,30,40,50]
>>> dir(l)
['append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop',
'remove', 'reverse', 'sort']
>>>
```

len():- Return the number of elements present in the list.

```
>>> len(l)
5
>>>
```

1. count():- It returns the number of occurance od specified item in the list.

```
>>> l = [1,1,2,2,2,3,4,4]
>>> l.count(3)
1
>>> l.count(4)
2
>>> l.count(1)
2
>>> l.count(2)
3
>>>
```

2. index():- Returns the index of the first occurance of the specified item.

```
>>> l = [1,1,2,2,2,3,4,4]
>>> l.index(1)
0
>>> l.index(3)
5
>>> l.index(4)
6
>>>
>>> l.index(6)
Traceback (most recent call last):
  File "<pyshell#21>", line 1, in <module>
    l.index(6)
ValueError: 6 is not in list
```

#Manipulating elements of list:-
----

3. append():- we can use append() to add item at the end of the list.

```
>>> l = [10,20,30]
>>> l.append(40)
>>> l
[10, 20, 30, 40]
>>> l.append(50)
>>> l
[10, 20, 30, 40, 50]
>>>
```

4. insert():- To insert at specified index position.

```
>>> l = [10,20,30]
>>> l.insert(1,15)
>>> l
[10, 15, 20, 30]
>>>
>>> l.insert(-1,35)
>>> l
[10, 15, 20, 35, 30]
```

difference between append() and insert():-

append()	insert
In list when we add any element it will come in last that is it will ve last element	In list we acn insert ane element in particular index number.

5. extend():- To add all items of one list to another list.

```
l1= []
l2 = []
```

l1.extend(l2):- l2 items will be added to l1

l2.extend(l1):- l1 items will be added to l2

```
>>> l = [10,20,30]
>>> p = [1,1,2,2,2,3,4,4]
>>>
>>> l.extend(p)
>>> l
[10, 20, 30, 1, 1, 2, 2, 2, 3, 4, 4]
>>>
```

6. remove():- we can use this function to remove specified item from the list.If the item present multiple times then only first occurance will be removed.

```
>>> l = [1,1,2,2,2,3,4,4]
>>> l.remove(3)
>>> l
[1, 1, 2, 2, 2, 4, 4]
>>> l.remove(1)
>>> l
[1, 2, 2, 2, 4, 4]
>>> l.remove(1)
>>> l
[2, 2, 2, 4, 4]
>>>

>>> l.remove(9)
Traceback (most recent call last):
  File "<pyshell#58>", line 1, in <module>
    l.remove(9)
ValueError: list.remove(x): x not in list
>>>
```

Note:- Hence before using remove() first we have to check specofied element present in the list or not by using in operator.

7. pop():- It removes and returns the last element of the list. This is only function which manipulates list and returns some element.

```
>>> l = [10,20,30]
>>> l.pop()
30
>>> l = [10,20,30,80,90,70]
>>> l.pop(2)
30
>>> l.pop(5)
Traceback (most recent call last):
  File "<pyshell#69>", line 1, in <module>
    l.pop(5)
IndexError: pop index out of range
>>> l.pop(-2)
90
>>>
```

Differenece betwen remoce() and pop()

remove()	pop()
1. We can use to remove special element from the list.	1. We can use to remove last element from the list.
2. It can't return any value.	2.It returned removed element.
3. If special element not available then we get ValueError.	3.If list is empty then get Error.

8.clear():- It's clear the all data from respective list.

```
>>> l = [10,20,30,80,90,70]
>>> l.clear()
>>> l
[]
```

9.copy():- copy the same data from a another list.

```
>> l = [10,20,30,80,90,70]
>>> a = l.copy()
>>> a
[10, 20, 30, 80, 90, 70]
>>>
```

10.reverse:- we can use to reverse() order of elemet list.

```
>>> l = [10,20,30,80,90,70]
>>> l.reverse()
>>> l
[70, 90, 80, 30, 20, 10]
>>>
```

11.sort():- In list by deafult insertion order is preserved.If want to sort the elements of list according to deafult natural sorting order then we should go for sort() function.

```
>>> l = [10,34,23,11.78,1,11,12.78,99,456,234]
>>> l.sort()
>>> l
[1, 10, 11, 11.78, 12.78, 23, 34, 99, 234, 456]
>>> l.sort(reverse=True)
>>> l
[456, 234, 99, 34, 23, 12.78, 11.78, 11, 10, 1]
>>>
```

```
>>> l = ['a','A','apple','ball','Apple','Ball']
>>> l.sort()
>>> l
['A', 'Apple', 'Ball', 'a', 'apple', 'ball']
```

```
l = [1,2,1.56,56,'A','Apple','a','b']
>>> l.sort()
Traceback (most recent call last):
  File "<pyshell#13>", line 1, in <module>
    l.sort()
TypeError: '<' not supported between instances of 'str' and 'int'
>>>
```