List Operator #1

- The operators on list discussed here are **[ ] , [:]**
- **[ ]** is used for **indexing** , Both positive and negative indexing is possible on list through indexing we can read the data of list as well as write / modify a list

```
>>>
>>> list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> list1[6]
7
>>> print(list1[6])
7
>>> print(list1[-4])
7
>>> x=list1[6]
>>> x
7
>>> list1[6]=15
>>> list1
[1, 2, 3, 4, 5, 6, 15, 8, 9, 10]
>>>
```

- **[:] - slice / slicing operator** is similar to index , but we can take a range of values in a list
- In slicing you can give **[start : end : stepsize]**
- Slicing will give you new list it doesn't modify the existing list
- It is used for reading either the complete list or a section of a list
- It is discussed in detail in the example below

```
>>>
>>> list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> list1[:]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> list1[3:]
[4, 5, 6, 7, 8, 9, 10]
>>> list1[3:8]
[4, 5, 6, 7, 8]
>>> list1[0:10]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> list1[0:9]
[1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> list1[0:10:2]
[1, 3, 5, 7, 9]
>>>
>>> temp = list1[0:10:2]
>>> list1
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> temp
[1, 3, 5, 7, 9]
>>> list1[::-1]
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
>>> list1[-1:-11:-1]
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
>>> list1[-1:-11:-2]
[10, 8, 6, 4, 2]
>>>
```

```
>>>
>>> list1 =[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> list1[0:3]=[10,20,30]
>>>
>>> list1
[10, 20, 30, 4, 5, 6, 7, 8, 9, 10]
>>> list1[0:3]=[11,12]
>>> list1
[11, 12, 4, 5, 6, 7, 8, 9, 10]
>>>
>>> list1[0:2]=[10,20,30,40,50]
>>> list1
[10, 20, 30, 40, 50, 4, 5, 6, 7, 8, 9, 10]
>>>
>>> list1[::2]=[11,22,33,44]
Traceback (most recent call last):
  File "<pyshell#11>", line 1, in <module>
    list1[::2]=[11,22,33,44]
ValueError: attempt to assign sequence of size 4 to extended slice of size 6
>>> list1[::2]=[11,22,33,44,55,66]
>>> list1
[11, 20, 22, 40, 33, 4, 44, 6, 55, 8, 66, 10]
>>>
```