29-9-2025 Monday

start value < stop value

Slicing :

↓[start position : stop position : step size]
var

* default start position is 0
* default stop position is end
* defaut step size is 1.

l = ['a', 2, 'e', 9.5, 'Besant', True, 'Tech', 34, 99.9]

⇒ To print elements from 'a' to True

l[0:6] o/p ['a', 2, 'e', 9.5, 'Besant', True

⇒ To print odd position elements

l[1::2] o/p [2, 9.5, True, 34]

⇒ To print all elements

l[:]

⇒ To print last 3 elements

l[-3:]

⇒ Print elements of list where position is divisible by 3

l[3::3] o/p [9.5, 'Tech']

⇒ Write a program to find the max elem from list without using max fn.

```
l = [4, 6, 1, 9, 2]
max_num = l4[0]
for i in l:
    if (i > max_num):
        max_num = i
print(max_num)
```

⇒ To sort in ascending/descending order.

* In ascending

~~l4.sort(reverse~~ l4.sort() l4

In descending

l4.sort(reverse = True) l4

⇒ Remove all the duplicates from list

l = [1, 2, 2, 3, 3, 3, 4, 4, 5, 6, 7, 8, 9, 9]

new_l = []

```
for i in l:
    if i not in new_l:
        new_l.append(i)
print(new_l)
```

o/p [1, 2, 3, 4, 5, 6, 7, 8, 9]

⇒ Remove all occurance of item 20

```
l = [5, 20, 15, 20, 25, 50, 20]
for i in l:
    if (i == 20):
        l.remove(20)
print(l)
```

o/p [5, 15, 25, 50]

⇒ Perform Union and Intersection on a given list

```
l1 = [1, 2, 3, 4, 5]
l2 = [4, 5, 6, 7, 8]
inter = []
for i in l1:
    if i in l2:
        inter.append(i)
print("Intersection:", inter)
uni = []
for i in l1+l2:
    if i not in uni:
        uni.append(i)
print("Union:", uni)
```

o/p Union : [1, 2, 3, 4, 5, 6, 7, 8,]
Intersection : [4, 5]

⇒ Remove empty string from the list of string

```
list = ["Arun", "", "Kamala", "", "John"]
new_list = []
for i in list:
    if (i != ""):
        new_list.append(i)
print(new_list)
```

o/p ['Arun', 'Kamala', 'John']

```
list = [ ]
for i in list:
    if (i == ""):
        list.remove(i)
print(list)
```

⇒ 2. Tuple: It is represented by tuple () @ ()

- It is ordered, immutable, allow duplicates.
- It is also heterogenous in nature

Ex `t = (28, 'Priya', 18.9, False, 'Karan')`

`t`

o/p (——————)