**>>** List **is** the collection of items

**>>** It **is** use to store sequence of data types

**>>** Data type:**-**

int,float,complex,bool,str,dict,set etc**...**

**>>** We can add any data type **in** list i.e. List **is** hetrogeneous data type

**>>** List **is** mutable (we can add,delete,update)

**>>** It **is** ordered data type (we can do slicing,indexing)

**>>** Duplicates are allowed

**>>** Enclosed by []

**>>** List items are comma separated

**>>** Tuple **is** immutable

**>>** It **is** ordered data type

**>>** It **is** enclosed by ()

**>>** Tuple items are comma separated

**>>** It allows duplicates

**>>** It has fixed length

**>>** It has less functionality

**>>** It required less memory

**>>** We can add multiple data types

**>>** Tuple **is** Faster than List

a **=** [1,2,1,3,5,1,8]

count **=** 0

n **=** 1

**for** i **in** a:

**if** i **==** n:

count **+=** 1

print(count)

lst1 **=** [1,2,3,4]

sum **=** 0

**for** i **in** lst1:

sum **+=** i

print(sum)

lst **=** [1,2,3,4,5,6,7,"samar"]

count **=** 0

**for** i **in** lst:

count **+=** 1

print(count)

1lst1 **=** [1,2,3,4]

2sum **=** 0

3**for** i **in** lst1:

4sum **+=** i

5print(sum)

10

1lst1 **=** [1,2,3]

2mult **=** 1

3**for** i **in** lst1:

4mult **\*=** i

5print(mult)

6

11) using len function

1lst **=** [1,2,3,4,5]

2len(lst)

Out[9]:

5

12) using loop

1lst **=** [1,2,3,4,5,6,7,"samar"]

2count **=** 0

3**for** i **in** lst:

4count **+=** 1

5print(count)

8

**Q12:- Python program to find the smallest and largest number ina list (Withoutmin-max function)**

a **=** [10,20,80,90,80,90,100]

b **=** sorted(a)[0]

print("Smallest nuber is:",b)

c **=** sorted(a)[**-**1]

print("Largest number is: ",c)

r **=** float(input("Enter the radius:"))

pi **=** 22**/**7

area\_circle **=** pi**\***r**\***r

print("Area of the circle is %2f" **%**area\_circle)

lst **=** []

n **=** int(input("Enter the length of the list:"))

**for** i **in** range(0,n):

e **=** int(input("Enter the element"))

lst.append(e)

print(lst)

p **=** int(input("Enter the no. to find the lst:"))

**if** p **in** lst:

lst.remove(p)

print(lst)

list\_even **=** []

list\_odd **=** []

**for** i **in** range(1,151):

**if** i**%**2 **==** 0:

list\_even.append(i)

**else**:

list\_odd.append(i)

print(f"Even list is = {list\_even}")

print("\*"**\***50)

print(f"Odd list is ={list\_odd}")

list\_even **=** [i **for** i **in** range(1,151) **if** i**%**2 **==** 0]

print("even list is = ",list\_even)

print("\*"**\***100)

list\_odd **=** [i **for** i **in** range(1,151) **if** i**%**2 **!=** 0]

print("Odd list is = ",list\_odd)

lst **=** [1,2,3,4,5,6,7,8,9,10]

even\_count **=** 0

odd\_count **=** 0

**for** i **in** lst:

**if** i**%**2 **==** 0:

even\_count **+=** 1

**else**:

odd\_count **+=** 1

print("even count is: ",even\_count)

print("odd count is: ",odd\_count)

even\_count **=** [i **for** i **in** range(1,11) **if** i**%**2 **==** 0]

print("even count is: ",len(even\_count))

odd\_count **=** [i **for** i **in** range(1,11) **if** i**%**2 **!=** 0]

print("Odd count is: ",len(odd\_count))

new\_lst **=** []

**for** i **in** range(0,151):

**if** i**%**4 **==** 0 **or** i**%**6 **==** 0 **or** i**%**10 **==** 0 **or** i**%**3 **==** 0 **or** i**%**5 **==** 0 **or** i**%**7 **==** 0 **or** i**%**9 **==**

new\_lst.append(i)

print(new\_lst)

lst **=** [1,3,5,"virat","rahul","samar",1.5,5.0,5.5]

int\_lst **=** []

float\_lst **=** []

str\_lst **=** []

**for** i **in** lst:

**if** type(i) **==** int:

int\_lst.append(i)

**elif** type(i) **==** float:

float\_lst.append(i)

**elif** type(i) **==** str:

str\_lst.append(i)

print(f"int list = {int\_lst}")

print(f"float list = {float\_lst}")

print(f"string list = {str\_lst}")

lst **=** [1,3,5,"virat","rahul","samar",1.5,5.0,5.5]

int\_lst **=** [i **for** i **in** lst **if** type(i) **==** int]

float\_lst **=** [i **for** i **in** lst **if** type(i) **==** float]

str\_lst **=** [i **for** i **in** lst **if** type(i) **==** str]

print(f"int list = {int\_lst}")

print(f"float list = {float\_lst}")

print(f"string list = {str\_lst}")

1) append()

**>>** It **is** use to add the items **and** items will get added at last index(**-**1)

**and** list will get updated itself.

*# example*

lst **=** [1,5,8,9]

lst.append(50)

lst

2) extend()

**>>** It **is** use to add the specific list items to the end of the current list **and** it will

a **=** [1,2,3]

b **=** [3,4,5]

a.extend(b)

lst1 **=** [1,2,3,4]

lst2 **=** [5,6,7,8]

lst1.append(lst2)

lst1 **=** [1,5,12,65,58,45]

a **=** sorted(lst1)[**-**3]

print("Third largest number is: ",a)

lst **=** ["A","B","C","F","S","a,","M"]

frequency **=** []

**for** i **in** lst:

**if** i **in** frequency:

frequency**+=**1

frequency.appendi

**else**:

frequrncy **=** 1

print(frequency)

lst **=** [1,2,3,2,3,5,7]

set1 **=** set(lst)

**for** i **in** set1:

print(f"{i} comes" ,lst.count(i),"times")

lst1 **=** ["virat","rahul",[1,2,3,4],10,5.5]

sublist **=** [1,2,3,4]

**if** sublist **in** lst1:

print(sublist)

**else**:

print("sublist not present")

lst **=** [1,2,3,4,5]

sub\_lst **=** [2,3]

print("Bigger list = ",lst)

print("Sublist is= ",sub\_lst)

**for** i **in** sub\_lst:

**if** i **in** lst:

print("sub\_lst is in Bigger lst")

**break**

**else**:

print("Sub\_lst is not in Bigger lst")

lst1 **=** []

sub\_lst **=** []

n **=** int(input("Enter the size of the lst:"))

print("Enter all element of the list:")

**for** i **in** range(n):

lst1.append(int(input("Enter the element to add: ")))

**for** i **in** range(len(lst)**+**1):

**for** j **in** range(i**+**1,len(lst1)**+**1):

sub\_lst.append(lst1[i:j])

print(lst1)

print(sub\_lst)

lst1 **=** [2,4,6,8]

lst2 **=** [1,4,5,8]

common\_lst **=** [i **for** i **in** lst1 **if** i **in** lst2]

print(common\_lst)

lst1 **=** [2,4,6,8]

lst2 **=** [1,4,5,8]

common\_lst **=** []

**for** i **in** lst1:

**if** i **in** lst2:

common\_lst.append(i)

print(common\_lst)

lst **=** [1,3,5,8,6,2,4] *# for ascending order*

**for** i **in** range(len(lst)):

**for** j **in** range(i**+**1,len(lst)):

**if** lst[i] **>** lst[j]:

lst[i],lst[j] **=** lst[j],lst[i]

print(lst)

\

lst **=** [1,3,5,8,6,2,4] *# disending order*

**for** i **in** range(len(lst)):

**for** j **in** range(i**+**1,len(lst)):

**if** lst[i] **<** lst[j]:

lst[i],lst[j] **=** lst[j],lst[i]

print(lst)

1. To sort a tuple we use sorted() function.

tup1 **=** (9,40,80,5)

b **=** sorted(tup1)

print(tuple(b))

**Q30:- Write a Python program to convert a list of multiple integersinto a single integer a. [11, 33, 50] >>> 113350**

lst **=** [11,33,50]

**for** i **in** lst:

print(i,end**=**"")

lst **=** [11,33,50]

lst2 **=** [str(i) **for** i **in** lst]

print("".join(lst2))

**-**The **del** keyword can delete the single value **from** a list by using index **or**

**-**The clear() method removes all the elements **from** a list.

a **=** [1,2,3,4]

**del** a[0]

a **=** [1,2,3,4]

a.clear()

a

Indexing :

It **is** used to obtain individual element.

Indexing can be positive **or** negative .

Slicing :

It **is** used to obtain a sequence of elements.Slicing returns a new list.

Slicing can be done by positive **and** negative indexing together

lst **=** [1,2,3]

lst[0]

lst **=** [1,2,3]

lst[0:2]

The sort() function returns nothing and changes the original sequence, while the sorted() function creates a new sequence type containing a sorted version of the given sequence.

a **=** [1,3,5,4]

sorted(a)

a **=** [1,3,5,4]

a.sort()

a **=** [1,2,3,4]

reversed(a)

a **=** [1,2,3,4]

a.reverse()

Copy :

No need to **import** it.

In case of nested loop,copy() function fails .

Copy() used to create the shallow copy of given list.

Deepcopy :

We have to **import** deepcopy **in** program

In case of nested loop deepcopy function works.

Deepcopy **is not** the list function

lst **=** [1,2,3,4]

**if not** lst:

print("List is Empty")

**else**:

print("List is not Empty")

a **=** [1,2,3]

b **=** [4,5,6]

a**+**b

a **=** [1,2,3]

b **=** [4,5,6]

a.extend(b)

a **=** [1,2,3]

b **=** [4,5,6]

a.append(b)

1) To concatenate two lst we use "+" operator

2) TO use extend() function

3) To use append() function

**Q39:- How to find the occurrences of an element in the pythonlist?**

a **=** [1,2,3,2,5]

a.count(2)

a **=** [1,2,3,4,5,2,3,2,8,2]

count **=** 0

n **=** 2

**for** i **in** a:

**if** i **==** n:

count **+=** 1

print(count)

a **=** [[1,2],[3,4],[4,5]] *# o/p - [1,2,3,4,5]*

b **=** []

**for** i **in** range(len(a)):

**for** j **in** range(len(a[i])):

b.append(a[i][j])

print(b)

a **=** [[1,2],[3,4],[4,5]]

b **=** []

**for** i **in** a:

**for** j **in** i:

b.append(j)

print(b)

a **=** [[1,2],[3,4],[4,5]]

b **=** [j **for** i **in** a **for** j **in** i]

print(b)