**List**

**List:**

* List is a collection of elements of similar or disimilar datatypes.
* Lists are 2 types.

1.empty list

2.non empty list

* Lists are denoted as square brackets[].
* List elements are seperated by comma(,).

Eg for creating list:

L=[1,2,3,’hai’,’23.6,5+6j]

* List elements are accessed by using index.

**List operations:**

* Slicing-it can be done with :
* Replication-it can be done with \*
* Adding-this can be done with +

Examples for list operations:

L1=[1,2,3]

L2=[4,5,6]

L1+L2--------------------🡪adding

L1[0:1]----------------------------🡪slicing

L1\*3----------🡪replication

**Deleting list elements:**

del(L1[2])

* lists are mutable.
* List contains both forward and reverse index.
* Forward index starts with 0 and reverse index starts with -1.
* List() method is used for type conversion.

Eg:s=’digit’

A=list(s)

Print(A)

The output is [‘d’,’i’,’g’,’i’,’t’]

* List has homogenous property.

**Accessing sublists:**

L=[1,2,3,4,[5,6]]

Print(L[4][1])

**List comprehension:**

* Creating list using loops.

Eg:

A=[i\*i for i in range(1,10)]

Print(A)

**Inbuilt functions in list:**

List()------type conversion

Len()------finding length

Sum()------finding sum

Min()------finding min element

Max()------finding max element

All()-------prints true if all the elements are high.

Any()--------print false if all the elements are 0.

**Inbuilt methods:**

**.append()**-----adding new sublist to the existing list.

**Eg**:print(l1.append(l2))

**.extend()---------**extending existing list.

**Eg:**s1.extend(s2)

**.**clear()--------------to delete list elements.

**Eg:**s.clear()

**.copy**():----------copying list from one to another.

S1=[1,2,3,4]

S2=[]

S2=s1.copy()

**.Count()------------**prints no of elements in a list.

**.insert() ----------**to insert a new element into an existing list.

**s.**insert(5,6)

here 5 is index and 6 is value.

**.remove**:

To remove an element from list.

s.remove(4)

**.pop:**to remove an element by passing index.

If we cant pass index it removes last element.

s.pop(3).

List works based on stack architecture.

**.reverse**: to reverse a given list.

**s**.reverse()

**sort**:

to arrange a list in an order.

**s**.sort()

**tuples**

* it is a collection of elements of similar or disimilar elements.
* Tuples are immutables.
* Tuples are represented in ( ).
* Tuples are 2 types.

1.packed-----written in parenthesis

2.unpacked-----written without using parenthesis.

**Declaring a tuple:**

T=(1,2,’hello’)

**Operations on tuples:**

* Slicing
* Reversing

**Functions in tuples:**

* Len
* Sum
* Max
* Min
* All
* Any
* Bool
* enumerate

**Converting tuple into list:**

Print(list(t))

**Tuple comprehension:**

A=(a for a in range(0,10))

Print(A)

It prints generator object because tuples are immutables.

**Getting data from generator object:**

For i in a

Print(i)

**Dictionaries**

* It is a collection of keys and values.
* These are mutable.
* These are iterable.
* Keys should be unique.
* Keys should be immutable objects.
* That means key maybe a number or sting or tuple.
* Dictionaries are denoted as { }.
* Keys and values are seperated by colon.

**Declaration of dictionary:**

D={1:’a’,2:’b’,3:’c’}

**Accessing dictionary elements:**

* Elements are accessed by using keys**.**
* **Eg:**
* D[1]

**Changing values:**

It can be done by using keys.

**Eg:**d[3]=’g’

**Deleting dictionary:**

Del d.

**Methods:**

**.keys**:prints list of keys**.**

**.values:**prints list of values.

**.items:** print both keys and values in the form of tuples**.**

**.popitem:**removes last item**.**

**.update:**adding items to dictionary**.**

* Dictionaries are arbitary.