Celebal :Week 3 :Document on Python

Data type:

1. Integer : 5 , 1049 ,23
2. Float : 5.6 , 0.5
3. Boolean: True , False
4. String: “Hello World”
5. Complex: 3+4i , 6-3i

Data Structure :

1. List : [1,2,3,4,5]
2. Tuple : (5,6,7,8)
3. Set : {1,2,3,6,8,9}
4. Dictionary :{ ‘Name’: “Sanjivani”, ‘Address’: “Kopargaon”}

Built-in Functions & Methods :

* List:

L =[1,5,3,7,9]

1. L.sort() :

L =[1,3,5,7,9]

Sort the element in ascending order.

1. L.reverse() :

L =[9,7,5,3,1]

Reverse the order.

1. L.remove(1):

L =[3,5,7,9]

Remove the element of the list.

1. L.append(10) :

L = [1,3,5,7,9,10]

The append() method appends an element to the end of the list.

1. L.max() :.

max = 9

Gives the maximum element

1. L.min() :

min = 1

Gives the minimum element

1. type() :

type() method returns class type of the argument(object) passed as parameter.

Type = list

* Tuples:
  + - T = (1,2,3,4,5,6,1,2,3,4,5,6)

1. count() : The count() method returns the number of times a specified value appears in the tuple.
2. index() : The index() method finds the first occurrence of the specified value. The index() method raises an exception if the value is not found.
3. len() : Python tuple method len() returns the number of elements in the tuple.

Lenth =12

1. max() : Python tuple method max() returns the elements from the tuple with maximum value.

Max = 6

1. min() : Python tuple method min() returns the elements from the tuple with minimum value.

Min = 1

1. tuple() : Python tuple method tuple() converts a list of items into tuples.

* Set:
  + - * set={1,2,3,4,5,6,2,2,2,3,3,3}

1. add() : The add() method adds an element to the set. If the element already exists, the add() method does not add the element.
2. clear() : The clear() method removes all elements from the set.
3. pop() : This in-built function of Python helps to pop out elements from a set just like the principle used in the concept while implementing Stack. This method removes a top element from the set but not the random element and returns the removed element.
4. union() : The union() method returns a set that contains all items from the original set, and all items from the specified set(s).
5. issubset() : The issubset() method returns True if all items in the set exists in the specified set, otherwise it retuns False.

* Dictionary:

dict=:{ ‘Name’: “Sanjivani”, ‘Address’: “Kopargaon”}

1. fromkeys() : The fromkeys() method returns a dictionary with the specified keys and the specified value.
2. get() : The get() method returns the value of the item with the specified key.
3. items() : The items() method returns a view object. The view object contains the key-value pairs of the dictionary, as tuples in a list.

|  |  |  |  |
| --- | --- | --- | --- |
| List | Tuple | Set | Dictionary |
| List is a non-homogeneous data structure that stores the elements in single row and multiple rows and columns | Tuple is also a non-homogeneous data structure that stores single row and multiple rows and columns | Set data structure is also non-homogeneous data structure but stores in single row | Dictionary is also a non-homogeneous data structure which stores key value pairs |
| List can be represented by  [ ] | Tuple can be represented by ( ) | Set can be represented by  { } | Dictionary  can be represented by { } |
| List allows duplicate elements | Tuple allows duplicate elements | Set will not allow duplicate elements | Set will not allow duplicate elements and dictionary doesn’t allow duplicate keys. |
| List can use nested among all | Tuple can use nested among all | Set can use nested among all | Dictionary can use nested among all |
| Example: [1, 2, 3, 4, 5] | Example: (1, 2, 3, 4, 5) | Example: {1, 2, 3, 4, 5} | Example: {1, 2, 3, 4, 5} |
| List can be created using list() function | Tuple can be created using tuple() function. | Set can be created using set() function | Dictionary can be created using dict() function. |
| List is mutable i.e we can make any changes in list. | Tuple  is immutable i.e we can not make any changes in tuple | Set is mutable i.e we can make any changes in set. But elements are not duplicated. | Dictionary is mutable. But Keys are not duplicated. |
| List is ordered | Tuple is ordered | Set is unordered | Dictionary is ordered (Python 3.7 and above) |
| Creating an empty list  l=[] | Creating an empty Tuple  t=() | Creating a set  a=set()  b=set(a) | Creating an empty dictionary  d={} |

**Difference Between** - List, Tuple, Set, Dictionary :

**Iterating For Loop:**

* A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string).
* This is less like the for keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.
* Ex:

Fruits=["apple", "banana", "cherry"]  
for x in fruits:  
  print(x)

Output:

apple

banana

cherry

**Membership Test:**

**1) In Operator:**

It checks whether the value is present in the sequence of data or not. It evaluates to **true** value if the element is present in the sequence and to **false** value if the element is absent from the sequence

Ex:

my\_new\_list = [1,2,3,'a']

# loop around the list

for i in my\_new\_list:

print(i)

output:

1

2

3

a

**2) not in operator:**

This operator checks for the absence of a value in a sequence. This is the exact opposite of the in operator. It evaluates to true when the element is not found or absent from the sequence and returns false when the element is found in the data sequence.

Ex:

my\_new\_list = [1,2,3, 'a']

special = 'Studytonight'

if special not in my\_new\_list:

print('True')

output:

True