15/1/2022

Tuple Data Type(tuple()):-

Tuple Data Type is exactly same as list data type except that is immutable ,we can not change values.

Tuple elements can be reprsented with in parenthsis.

```
s = (1,2,3,4)
type(s)
<class 'tuple'>
```

```
s = (1,2,3,4)
dir(s)
[count', 'index']
```

Note:- tuple is the read only version of list.

Set Data type(set()):-

If we want to reprsent a group of values without duplicates where order is not important then we should go for set data type.

ex:-

```
s = {1,1,2,2,2,3,4,'scodeen',True,(78,90)}
s
{1, 2, 3, 4, (78, 90), 'scodeen'}
type(s)
<class 'set'>
```

Notes:-

1.Insertion order is not preserved.
2.Duplicates are not allowed.
3.hetrogeneous object are allowed.
4.Index concept is not applicable.
5.Set is mutable.
6.Growable in nature.

set manipulation:-

```
s = {1,2,3,4,5,6,7}
dir(s)
['add', 'clear', 'copy', 'difference', 'difference_update', 'discard', 'intersection', 'intersection_update',
'isdisjoint', 'issubset', 'issuperset', 'pop', 'remove', 'symmetric_difference',
'symmetric_difference_update', 'union', 'update']
```

imp:- 'add', 'clear', 'copy', 'difference','discard', 'intersection', 'pop', 'remove','union', 'update']

1.add():- Add itmes x to the set.

```
s = {1,2,3,4,5,6,7}
s.add(8)
s
{1, 2, 3, 4, 5, 6, 7, 8}
```

2.clear():- To remove all the elements from the set,

```
s = {1,2,3,4,5,6,7}
s.clear()
s
set()
```

3.copy():- Create a mirror image in other variable.

```
s = {1,2,3,4,5,6,7}
a = s.copy()
a
{1, 2, 3, 4, 5, 6, 7}
type(a)
<class 'set'>
```

4.differenece():- Returns the elements present in x but not in y.

```
x = {10,20,30,40,50}
y = {30,40,50,60}
a = x.difference(y)
a
{10, 20}
b = y.difference(x)
b
{60}
```

5.discard():- It removes the specified elements from the set.
If the specified element not present in the set then we won'get any error.

```
x = {10,20,30,40,50}
x.discard(40)
x
{50, 20, 10, 30}
x.discard(800)
x
{50, 20, 10, 30}
```

6.pop():- It removes and returns some random elemts from the set.

```
x = {10,20,30,40,50}
x.pop()
50
x.pop()
20
```

7,remove():-

It remove the specified element from the set.
If the specofied element not present in the set then we will get error.

```
x = {10,20,30,40,50}
x.remove(30)
x
{50, 20, 40, 10}
x.remove(300)
Traceback (most recent call last):
File "<pyshell#49>", line 1, in <module>
x.remove(300)
KeyError: 300
```

8.intersection():-

Return common elements present in both the set.

x = {10,20,30,40,50}

```
y = {30,40,50,60}

x.intersection(y)
{40, 50, 30}
y.intersection(x)
{40, 50, 30}
```

9.union(): We can use this function to return all elements present in both set.
x.union(y) or y.union(x)

```
x = {10,20,30,40}
y = {30,40,50,60}
x.union(y)
{40, 10, 50, 20, 60, 30}
```

10.update():-
To add multiple items to the set.

```
s = {10,20,30}
t = (67,78,89)
s.update(t)
s
{67, 10, 78, 20, 89, 30}
```

Frozenset data type(frozenset()):-

s = {10,20,30}

```
fs = frozenset(s)
fs
frozenset({10, 20, 30})
type(fs)
<class 'frozenset'>
```

```
dir(fs)
[ 'copy', 'difference', 'intersection', 'isdisjoint', 'issubset', 'issuperset', 'symmetric_difference', 'union']
```