- List
- Tuple
- Set
- Dict
- Range

LIST

In [1]:
```
l = []
l
```

Out[1]:
```
[]
```

In [2]:
```
type(l)
```

Out[2]:
```
list
```

In [3]:
```
len(l)
```

Out[3]:
```
0
```

In [4]:
```
l.append()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[4], line 1
----> 1 l.append()

TypeError: list.append() takes exactly one argument (0 given)
```

In [5]:
```
l.append(10)
```

In [6]:
```
l
```

Out[6]:
```
[10]
```

In [7]:
```
l.append(20)
l.append(30)
l.append(40)
```

In [8]:
```
l
```

Out[8]:
```
[10, 20, 30, 40]
```

In [9]:
```
len(l)
```

Out[9]:
```
4
```

In [10]:
```
l.append(50)
```

In [11]:
```
l
```

Out[11]:
```
[10, 20, 30, 40, 50]
```

In [12]:
```
l1 = [10, 3.4, 'ten', True, 1+2j]
l1
```

Out[12]:
```
[10, 3.4, 'ten', True, (1+2j)]
```

In [13]:
```
print(l)
print(l1)
```

```
[10, 20, 30, 40, 50]
[10, 3.4, 'ten', True, (1+2j)]
```

In [14]:
```
print(len(l))
print(len(l1))
```

```
5
5
```

In [15]:
```
print(id(l))
print(id(l1))
```

```
2845755717760
2845770434688
```

In [16]:
```
l2
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[16], line 1
----> 1 l2

NameError: name 'l2' is not defined
```

In [17]:
```
l2 = l1.copy()
l2
```

Out[17]:
```
[10, 3.4, 'ten', True, (1+2j)]
```

In [18]:
```
l1 == l2
```

Out[18]:
```
True
```

In [19]:
```
l == l2
```

Out[19]:
```
False
```

In [20]:
```
print(l)
print(l1)
print(l2)
```

```
[10, 20, 30, 40, 50]
[10, 3.4, 'ten', True, (1+2j)]
[10, 3.4, 'ten', True, (1+2j)]
```

In [21]:
```
l != l2
```

Out[21]:
```
True
```

In [22]:
```
l
```

Out[22]:
```
[10, 20, 30, 40, 50]
```

In [23]:
```
l[:]
```

Out[23]:
```
[10, 20, 30, 40, 50]
```

In [24]:
```
l[0]
```

Out[24]:
```
10
```

In [25]:
```
l[-1]
```

Out[25]:
```
50
```

In [26]:
```
l
```

Out[26]:
```
[10, 20, 30, 40, 50]
```

In [27]:
```
l[4]
```

Out[27]:
```
50
```

In [28]:
```
l[5]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[28], line 1
----> 1 l[5]

IndexError: list index out of range
```

In [29]:
```
l[10]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[29], line 1
----> 1 l[10]

IndexError: list index out of range
```

In [30]:
```
l
```

Out[30]:
```
[10, 20, 30, 40, 50]
```

In [31]:
```
l[0]
```

Out[31]:
```
10
```

In [32]:
```
l[0] = 100
```

In [33]:
```
l
```

Out[33]:
```
[100, 20, 30, 40, 50]
```

In [34]:
```
l[-1] = 'jan'
```

In [35]:
```
l
```

Out[35]:
```
[100, 20, 30, 40, 'jan']
```

In [36]:
```
l.append('feb')
```

In [37]:
```
l
```

Out[37]:
```
[100, 20, 30, 40, 'jan', 'feb']
```

In [38]:
```
l.append(40)
l
```

Out[38]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [39]:
```
l
```

Out[39]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [40]:
```
l2
```

Out[40]:
```
[10, 3.4, 'ten', True, (1+2j)]
```

In [41]:
```
len(l2)
```

Out[41]:
```
5
```

In [42]:
```
l2.clear()
```

In [43]:
```
l2
```

Out[43]:
```
[]
```

In [44]:
```
len(l2)
```

Out[44]:
```
0
```

In [45]:
```
del l2
```

In [46]:
```
l
```

Out[46]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [47]:
```
l.count(100)
```

Out[47]:
```
1
```

In [48]:
```
l.count(40)
```

Out[48]:
```
2
```

In [49]:
```
l
```

Out[49]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [50]:
```
l.count(1000)
```

Out[50]:
```
0
```

In [51]:
```
print(l)
print(l1)
```

```
[100, 20, 30, 40, 'jan', 'feb', 40]
[10, 3.4, 'ten', True, (1+2j)]
```

In [52]:
```
print(len(l))
print(len(l1))
```

```
7
5
```

list membership

In [53]:
```
l
```

Out[53]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [54]:
```
100 in l
```

Out[54]:
```
True
```

In [55]:
```
1000 in l
```

Out[55]:
```
False
```

In [56]:
```
l
```

Out[56]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [57]:
```
l1
```

Out[57]: [10, 3.4, 'ten', True, (1+2j)]

In [58]:
```
l2 = []
```

In [59]:
```
print(len(l))
print(len(l1))
print(len(l2))
```

```
7
5
0
```

In [60]:
```
l
```

Out[60]: [100, 20, 30, 40, 'jan', 'feb', 40]

In [61]:
```
l1
```

Out[61]: [10, 3.4, 'ten', True, (1+2j)]

In [62]:
```
l2
```

Out[62]: []

In [63]:
```
l2.extend(l)
```

In [64]:
```
l2
```

Out[64]: [100, 20, 30, 40, 'jan', 'feb', 40]

In [65]:
```
l2.extend(l1)
```

In [66]:
```
l2
```

Out[66]: [100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]

In [67]:
```
l1
```

Out[67]: [10, 3.4, 'ten', True, (1+2j)]

In [68]:
```
l1.extend(l2)
```

In [69]:
```
l1
```

Out[69]:
```
[10,
 3.4,
 'ten',
 True,
 (1+2j),
 100,
 20,
 30,
 40,
 'jan',
 'feb',
 40,
 10,
 3.4,
 'ten',
 True,
 (1+2j)]
```

In [70]:
```
len(l1)
```

Out[70]:
```
17
```

In [71]:
```
l
```

Out[71]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [72]:
```
l.index('jan')
```

Out[72]:
```
4
```

In [73]:
```
l
```

Out[73]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [74]:
```
l[:]
```

Out[74]:
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [75]:
```
l2
```

Out[75]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [76]:
```
l2[:]
```

Out[76]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [77]:
```
l2[0:6]
```

Out[77]:
```
[100, 20, 30, 40, 'jan', 'feb']
```

In [78]:
```
l2
```

Out[78]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [79]:
```
l2[3:10]
```

Out[79]:
```
[40, 'jan', 'feb', 40, 10, 3.4, 'ten']
```

In [80]:
```
l2
```

Out[80]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [81]:
```
l2[::2]
```

Out[81]:
```
[100, 30, 'jan', 40, 3.4, True]
```

In [82]:
```
l2
```

Out[82]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [83]:
```
l2[::5]
```

Out[83]:
```
[100, 'feb', True]
```

backwaard indexing & backward slicing

In [84]:
```
l2
```

Out[84]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [85]:
```
l2[:]
```

Out[85]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [86]:
```
l2[-5]
```

Out[86]:
```
10
```

In [87]:
```
l2
```

Out[87]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [88]:
```
l2[:-9]
```

Out[88]:
```
[100, 20, 30]
```

In [89]:
```
l2
```

Out[89]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [90]:
```
l2[:-3]
```

Out[90]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4]
```

In [91]:
```
l2
```
Out[91]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [92]:
```
l2[-3:]
```
Out[92]:
```
['ten', True, (1+2j)]
```

In [93]:
```
l2
```
Out[93]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [94]:
```
l2[3:-3]
```
Out[94]:
```
[40, 'jan', 'feb', 40, 10, 3.4]
```

In [95]:
```
l2
```
Out[95]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [96]:
```
l2[::-1]
```
Out[96]:
```
[(1+2j), True, 'ten', 3.4, 10, 40, 'feb', 'jan', 40, 30, 20, 100]
```

In [97]:
```
l2
```
Out[97]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [98]:
```
l2[:-3]
```
Out[98]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4]
```

In [99]:
```
l2
```
Out[99]:
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [100…
```
l2[::-3]
```
Out[100…
```
[(1+2j), 3.4, 'feb', 30]
```

In [101…
```
print(l)
print(l1)
print(l2)
```
```
[100, 20, 30, 40, 'jan', 'feb', 40]
[10, 3.4, 'ten', True, (1+2j), 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', Tr
ue, (1+2j)]
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [102…
```
l1.remove(1+2j)
```

In [103…
```
print(l1)
```

```
[10, 3.4, 'ten', True, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2
j)]
```

23rd

In [104…
```
print(l)
print(l1)
print(l2)
```

```
[100, 20, 30, 40, 'jan', 'feb', 40]
[10, 3.4, 'ten', True, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2
j)]
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2j)]
```

In [106…
```
l
```

Out[106…
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [109…
```
l.append(10)
```

In [110…
```
l
```

Out[110…
```
[100, 20, 30, 40, 'jan', 'feb', 40, 10, 10]
```

In [115…
```
l.remove(10)
```

In [116…
```
l
```

Out[116…
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [117…
```
l
```

Out[117…
```
[100, 20, 30, 40, 'jan', 'feb', 40]
```

In [118…
```
l.insert(25,2)
```

In [119…
```
l
```

Out[119…
```
[100, 20, 30, 40, 'jan', 'feb', 40, 2]
```

In [120…
```
l.insert(2,25)
```

In [121…
```
l
```

Out[121…
```
[100, 20, 25, 30, 40, 'jan', 'feb', 40, 2]
```

In [122…
```
l.insert(4, 35)
```

In [123…
```
l
```

Out[123…
```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40, 2]
```

In [124…
```
l.insert([1,2,3], 10)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[124], line 1
----> 1 l.insert([1,2,3], 10)

TypeError: 'list' object cannot be interpreted as an integer
```

In [126…
```
l.append([1,2,3])
l
```

Out[126…
```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40, 2, [1, 2, 3]]
```

In [127…
```
l
```

Out[127…
```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40, 2, [1, 2, 3]]
```

In [128…
```
l.pop()
```

Out[128…
```
[1, 2, 3]
```

In [129…
```
l
```

Out[129…
```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40, 2]
```

In [130…
```
l.pop()
```

Out[130…
```
2
```

In [131…
```
l
```

Out[131…
```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40]
```

In [133…
```
print(l1)
```

```
[10, 3.4, 'ten', True, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True, (1+2
j)]
```

In [135…
```
l1.remove((1+2j))
```

In [136…
```
l1
```

Out[136…
```
[10, 3.4, 'ten', True, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True]
```

In [137…
```
l1.remove('ten')
```

In [138…
```
l1
```

Out[138…
```
[10, 3.4, True, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True]
```

In [140…
```
l1.pop(True)
```

Out[140… True

In [142…
```
l1
```

Out[142… [10, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True]

In [143…
```
l1
```

Out[143… [10, 100, 20, 30, 40, 'jan', 'feb', 40, 10, 3.4, 'ten', True]

In [144…
```
l1.pop('jan')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[144], line 1
----> 1 l1.pop('jan')

TypeError: 'str' object cannot be interpreted as an integer
```

In [145…
```
l1.remove('jan')
```

In [146…
```
l1
```

Out[146… [10, 100, 20, 30, 40, 'feb', 40, 10, 3.4, 'ten', True]

In [147…
```
l1.pop(5)
```

Out[147… 'feb'

In [148…
```
l1
```

Out[148… [10, 100, 20, 30, 40, 40, 10, 3.4, 'ten', True]

In [149…
```
l3 = [100, 2, 39,34, 20,1000]
l3
```

Out[149… [100, 2, 39, 34, 20, 1000]

In [152…
```
l3.sort(reverse=True) # hyperparameter
```

In [153…
```
l3
```

Out[153… [1000, 100, 39, 34, 20, 2]

In [154…
```
l3.sort()  # parameter
```

In [155…
```
l3
```

Out[155… [2, 20, 34, 39, 100, 1000]

In [156…
```
l4 = [2, 2.3, True, 'ten', 1+2j]
l4
```

Out[156… [2, 2.3, True, 'ten', (1+2j)]

In [157…
```
l4.sort()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[157], line 1
----> 1 l4.sort()

TypeError: '<' not supported between instances of 'str' and 'int'
```

In [159…
```
l5 = ['m', 'b', 'z', 'a']
l5
```

Out[159… ['m', 'b', 'z', 'a']

In [160…
```
l5.sort()
```

In [161…
```
l5
```

Out[161… ['a', 'b', 'm', 'z']

In [162…
```
l5.sort(reverse=True)
```

In [163…
```
l5
```

Out[163… ['z', 'm', 'b', 'a']

In [164…
```
l1
```

Out[164… [10, 100, 20, 30, 40, 40, 10, 3.4, 'ten', True]

In [165…
```
l1.reverse()
```

In [166…
```
l1
```

Out[166… [True, 'ten', 3.4, 10, 40, 40, 30, 20, 100, 10]

In [168…
```
l
```

Out[168… [100, 20, 25, 30, 35, 40, 'jan', 'feb', 40]

In [173…
```
print(l[0])
print(l[1])
print(l[2])
print(l[3])
print(l[4])
print(l[5])
print(l[6])
print(l[7])
print(l[8])
print(l[9])
```

```
100
20
25
30
35
40
jan
feb
40
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[173], line 10
      8 print(l[7])
      9 print(l[8])
---> 10 print(l[9])

IndexError: list index out of range
```

In [174…

```
l
```

Out[174…

```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40]
```

In [175…

```
for i in l:
    for j in l1:

    print(i)
```

```
100
20
25
30
35
40
jan
feb
40
```

In [176…

```
l
```

Out[176…

```
[100, 20, 25, 30, 35, 40, 'jan', 'feb', 40]
```

In [177…

```
l[6]
```

Out[177…

```
'jan'
```

In [178…

```
print(l[6][0])
print(l[6][1])
print(l[6][2])
```

```
j
a
n
```

In [182…

```
l3
```

Out[182… [2, 20, 34, 39, 100, 1000]

In [183…
```
for i in l3:
    print(i)
```

```
2
20
34
39
100
1000
```

In [184…
```
for i in enumerate(l3):
    print(i)
```

```
(0, 2)
(1, 20)
(2, 34)
(3, 39)
(4, 100)
(5, 1000)
```

In [186…
```
L1 = [1,2,3,4,0]
```

In [187…
```
all(L1)
```

Out[187… False

In [188…
```
any(L1)
```

Out[188… True

In [189…
```
L2 = [1,2,3,4]
```

In [190…
```
all(L2)
```

Out[190… True

In [191…
```
any(L2)
```

Out[191… True

tuple

In [2]:
```
t = ()
t
```

Out[2]: ()

In [3]:
```
type(t)
```

Out[3]: tuple

In [4]:
```
t = (10,20,30)
t
```

Out[4]: (10, 20, 30)

In [6]:
```
t.count(20)
```

Out[6]: 1

In [9]:
```
t1 = (10,20,2.2,'ten', True, 1+2j,20)
t1
```

Out[9]: (10, 20, 2.2, 'ten', True, (1+2j), 20)

In [10]:
```
t1.count(20)
```

Out[10]: 2

In [11]:
```
t1.index(20)
```

Out[11]: 1

In [12]:
```
print(t)
print(t1)
```

```
(10, 20, 30)
(10, 20, 2.2, 'ten', True, (1+2j), 20)
```

In [14]:
```
print(len(t))
print(len(t1))
```

```
3
7
```

In [15]:
```
t
```

Out[15]: (10, 20, 30)

In [16]:
```
t[0]
```

Out[16]: 10

In [17]:
```
t[0] = 100
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[17], line 1
----> 1 t[0] = 100

TypeError: 'tuple' object does not support item assignment
```

In [18]:
```
bank_account = (1234, 'cizp45yi', 10000)
bank_account
```

Out[18]:
```
(1234, 'cizp45yi', 10000)
```

In [19]:
```
bank_account[2] = 200000
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[19], line 1
----> 1 bank_account[2] = 200000

TypeError: 'tuple' object does not support item assignment
```

In [20]:
```
t
```

Out[20]:
```
(10, 20, 30)
```

In [21]:
```
t2 = t * 3
t2
```

Out[21]:
```
(10, 20, 30, 10, 20, 30, 10, 20, 30)
```

In [25]:
```
t
```

Out[25]:
```
(10, 20, 30)
```

In [26]:
```
for i in t:
    print(i)
```

```
10
20
30
```

In [27]:
```
for i in enumerate(t):
    print(i)
```

```
(0, 10)
(1, 20)
(2, 30)
```

list

- mutable
- duplicate is allowed
- append(), copy(), insert(), extend(), pop()
- remove the element
- list is growable
- multiple data type in a list
- indexing & slicing is allowed

Tuple

- immutable (unchangeable)
- duplication is allowed
- remove is not allowed
- only 2 function will work (.index, .count)

Set

In [28]:
```
s = {}
s
```

Out[28]:
```
{}
```

In [29]:
```
type(s)
```

Out[29]:
```
dict
```

In [30]:
```
s1 = set()
s1
```

Out[30]:
```
set()
```

In [31]:
```
s2 = {90, 10, 50, 40, 25, 10, 50 }
s2
```

Out[31]:
```
{10, 25, 40, 50, 90}
```

In [32]:
```
type(s2)
```

Out[32]:
```
set
```

In [33]:
```
s2
```

Out[33]:
```
{10, 25, 40, 50, 90}
```

In [35]:
```
s3 = s2.copy()
s3
```

Out[35]:
```
{10, 25, 40, 50, 90}
```

In [36]:
```
s3
```

Out[36]:
```
{10, 25, 40, 50, 90}
```

In [37]:
```
s3.add(3.4)
```

In [38]:
```
s3
```

Out[38]: {3.4, 10, 25, 40, 50, 90}

In [40]:
```
s3.add('nit')
```

In [41]:
```
s3
```

Out[41]: {10, 25, 3.4, 40, 50, 90, 'nit'}

In [42]:
```
s3.add(1+2j)
s3.add(True)
```

In [44]:
```
s3
```

Out[44]: {(1+2j), 10, 25, 3.4, 40, 50, 90, True, 'nit'}

In [46]:
```
print(s)
print(s1)
print(s2)
print(s3)
```

```
{}
set()
{50, 90, 40, 25, 10}
{True, 3.4, (1+2j), 10, 25, 90, 40, 'nit', 50}
```

In [47]:
```
s
```

Out[47]: {}

In [51]:
```
type(s)
```

Out[51]: dict

In [52]:
```
s2
```

Out[52]: {10, 25, 40, 50, 90}

In [53]:
```
s3
```

Out[53]: {(1+2j), 10, 25, 3.4, 40, 50, 90, True, 'nit'}

In [55]:
```
s3.remove(1+2j)
```

In [56]:
```
s3
```

Out[56]: {10, 25, 3.4, 40, 50, 90, True, 'nit'}

In [57]:
```
s3.remove(2000)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[57], line 1
----> 1 s3.remove(2000)

KeyError: 2000
```

In [58]:
```
s3
```

Out[58]:
```
{10, 25, 3.4, 40, 50, 90, True, 'nit'}
```

In [60]:
```
s3.discard(10)
```

In [61]:
```
s3
```

Out[61]:
```
{25, 3.4, 40, 50, 90, True, 'nit'}
```

In [62]:
```
s3.discard(2000)
```

In [63]:
```
s3
```

Out[63]:
```
{25, 3.4, 40, 50, 90, True, 'nit'}
```

In [64]:
```
s3.pop()
```

Out[64]:
```
True
```

In [65]:
```
s3
```

Out[65]:
```
{25, 3.4, 40, 50, 90, 'nit'}
```

In [66]:
```
s3.pop()
```

Out[66]:
```
3.4
```

In [67]:
```
s3
```

Out[67]:
```
{25, 40, 50, 90, 'nit'}
```

In [68]:
```
s3.pop(0)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[68], line 1
----> 1 s3.pop(0)

TypeError: set.pop() takes no arguments (1 given)
```

In [69]:
```
s3
```

Out[69]:
```
{25, 40, 50, 90, 'nit'}
```

In [70]:
```
s3[:]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[70], line 1
----> 1 s3[:]

TypeError: 'set' object is not subscriptable
```

In [71]:
```
s3
```

Out[71]:
```
{25, 40, 50, 90, 'nit'}
```

In [72]:
```
s3[1:]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[72], line 1
----> 1 s3[1:]

TypeError: 'set' object is not subscriptable
```

In [73]:
```
s3
```

Out[73]:
```
{25, 40, 50, 90, 'nit'}
```

In [74]:
```
s3[2]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[74], line 1
----> 1 s3[2]

TypeError: 'set' object is not subscriptable
```

In [75]:
```
s3
```

Out[75]:
```
{25, 40, 50, 90, 'nit'}
```

In [76]:
```
s3.pop(0)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[76], line 1
----> 1 s3.pop(0)

TypeError: set.pop() takes no arguments (1 given)
```

In [77]:
```
s3.pop()
```

Out[77]:
```
25
```

In [78]:
```
s3
```

Out[78]:
```
{40, 50, 90, 'nit'}
```

In [79]:
```
40 in s3
```

Out[79]:
```
True
```

set operation

In [80]:
```
a = {1,2,3,4,5}
b = {4,5,6,7,8}
c = {8,9,10}
```

In [83]:
```
type(c)
```

Out[83]:
```
set
```

In [84]:
```
a.union(b)
```

Out[84]:
```
{1, 2, 3, 4, 5, 6, 7, 8}
```

In [85]:
```
a.union(b, c)
```

Out[85]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In [86]:
```
print(a)
print(b)
print(c)
```

```
{1, 2, 3, 4, 5}
{4, 5, 6, 7, 8}
{8, 9, 10}
```

In [87]:
```
a | b
```

Out[87]:
```
{1, 2, 3, 4, 5, 6, 7, 8}
```

In [88]:
```
b | c
```

Out[88]:
```
{4, 5, 6, 7, 8, 9, 10}
```

In [89]:
```
a | b | c
```

Out[89]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In [90]:
```
a | c
```

Out[90]:
```
{1, 2, 3, 4, 5, 8, 9, 10}
```

In [91]:
```
a | c | b
```

Out[91]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

intersection

In [93]:
```
a = {1,2,3,4,5}
b = {4,5,6,7,8}
c = {8,9,10}
```

In [94]:
```
a.intersection(b)
```

Out[94]:
```
{4, 5}
```

In [95]:
```
b.intersection(c)
```

Out[95]:
```
{8}
```

In [96]:
```
a & b
```

Out[96]:
```
{4, 5}
```

In [97]:
```
b & c
```

Out[97]:
```
{8}
```

Differnece

In [98]:
```
a = {1,2,3,4,5}
b = {4,5,6,7,8}
c = {8,9,10}
```

In [99]:
```
a.difference(b)
```

Out[99]:
```
{1, 2, 3}
```

In [100…
```
b.difference(a)
```

Out[100…
```
{6, 7, 8}
```

In [ ]:
```
a = {1,2,3,4,5}
b = {4,5,6,7,8}
c = {8,9,10}
```

In [101…
```
b - c
```

Out[101…
```
{4, 5, 6, 7}
```

In [102…
```
c - b
```

Out[102…
```
{9, 10}
```

In [103…
```
a - b - c
```

Out[103…
```
{1, 2, 3}
```

In [ ]:
```

```