Sets

```
In [1]: myset={1,2,3,4,5}
        myset

Out[1]: {1, 2, 3, 4, 5}

In [2]: len(myset)

Out[2]: 5

In [3]: my_set={1,1,2,2,3,4,5,5}
        my_set

Out[3]: {1, 2, 3, 4, 5}

In [4]: myset1={1.79,2.08,3.99,4.56,5.45}
        myset1

Out[4]: {1.79, 2.08, 3.99, 4.56, 5.45}

In [5]: myset2={'Asif','John','Tyrion'}
        myset2

Out[5]: {'Asif', 'John', 'Tyrion'}

In [6]: myset3={10,20,"Hola",(11,22,32)}
        myset3

Out[6]: {(11, 22, 32), 10, 20, 'Hola'}

In [7]: myset3={10,20,"Hola",[11,22,32]}
        myset3
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[7], line 1
----> 1 myset3={10,20,"Hola",[11,22,32]}
      2 myset3

TypeError: unhashable type: 'list'
```

```
In [8]: myset4=set()
        print(type(myset4))

<class 'set'>

In [9]: type(set4)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[9], line 1
----> 1 type(set4)

NameError: name 'set4' is not defined
```

```
In [10]: type(myset4)

Out[10]: set

In [11]: my=set(1,3)
         my
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[11], line 1
----> 1 my=set(1,3)
      2 my

TypeError: set expected at most 1 argument, got 2
```

```
In [12]: my=set(2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[12], line 1
----> 1 my=set(2)

TypeError: 'int' object is not iterable
```

```
In [13]: my=set()
         print(type(my))

<class 'set'>

In [14]: my_set1=set(('one','two','three','four'))
         my_set1

Out[14]: {'four', 'one', 'three', 'two'}

In [15]: type(my_set1)

Out[15]: set

In [16]: my=set('22')
         type(my)

Out[16]: set
```

loop through set

```
In [18]: myset={'one','two','three','four','five','six','seven','eight'}
         for i in myset:
             print(i)

one
three
six
five
seven
two
four
eight
```

In [19]:

```
for i in enumerate(myset):
    print(i)
```

```
(0, 'one')
(1, 'three')
(2, 'six')
(3, 'five')
(4, 'seven')
(5, 'two')
(6, 'four')
(7, 'eight')
```

In [20]:

```
if 'one' in myset:
    print("one is present in myset")
else:
    print("one is not present in myset")
```

```
one is present in myset
```

In [21]:

```
myset
```

Out[21]:

```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [22]:

```
'one' in myset
```

Out[22]:

```
True
```

In [23]:

```
'ten' in myset
```

Out[23]:

```
False
```

In [25]:

```
if 'three' in myset:
    print("Three is present in myset")
else:
    print("Three is not present in myset")
```

```
Three is present in myset
```

In [26]:

```
if 'eleven' in myset:
    print("Eleven is present in myset")
else:
    print("Eleven is not present in myset")
```

```
Eleven is not present in myset
```

Add & Remove Items

In [27]:

```
myset
```

Out[27]:

```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [30]:

```
myset.add('Nine')
myset
```

```
Out[30]: {'Nine',
          'eight',
          'five',
          'four',
          'nine',
          'one',
          'seven',
          'six',
          'three',
          'two'}
```

```
In [31]: myset('9')
         myset
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[31], line 1
----> 1 myset('9')
      2 myset

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

```
In [32]: myset.add('9')
         myset
```

```
Out[32]: {'9',
          'Nine',
          'eight',
          'five',
          'four',
          'nine',
          'one',
          'seven',
          'six',
          'three',
          'two'}
```

```
In [33]: myset.remove('9')
         myset
```

```
Out[33]: {'Nine',
          'eight',
          'five',
          'four',
          'nine',
          'one',
          'seven',
          'six',
          'three',
          'two'}
```

```
In [34]: myset.update(['TEN','ELEVEN','TWELVE'])
         myset
```

```
Out[34]: {'ELEVEN',
           'Nine',
           'TEN',
           'TWELVE',
           'eight',
           'five',
           'four',
           'nine',
           'one',
           'seven',
           'six',
           'three',
           'two'}
```

```
In [35]: myset.remove('Nine')
         myset
```

```
Out[35]: {'ELEVEN',
           'TEN',
           'TWELVE',
           'eight',
           'five',
           'four',
           'nine',
           'one',
           'seven',
           'six',
           'three',
           'two'}
```

```
In [39]: myset.discard('TEN')
         myset
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[39], line 1
----> 1 myset.discard('TEN')
      2 myset

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

```
In [37]: myset.clear()
         myset
```

```
Out[37]: set()
```

```
In [38]: del myset
         myset
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[38], line 2
      1 del myset
----> 2 myset

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

```
In [40]: myset={1,3,4}
         myset
```

```
Out[40]: {1, 3, 4}

In [41]: myset={'one','two','three','four','five','six','seven','eight'}
         myset

Out[41]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [42]: myset1=myset
         myset1

Out[42]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [43]: id(myset),id(myset1)

Out[43]: (2268504198720, 2268504198720)

In [44]: my_set=mysset.copy()
         my_set

       ---------------------------------------------------------------------------
       NameError                                 Traceback (most recent call last)
       Cell In[44], line 1
       ----> 1 my_set=mysset.copy()
             2 my_set

       NameError: name 'mysset' is not defined

In [45]: my_set=myset.copy()
         my_set

Out[45]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [46]: my_set=myset1.copy()
         my_set

Out[46]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [47]: id(my_set)

Out[47]: 2268504357152

In [48]: myset.add('nine')
         myset

Out[48]: {'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}

In [49]: myset1

Out[49]: {'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}

In [50]: my_set

Out[50]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

set operation

In [51]:
```
A={1,2,3,4,5}
B={4,5,6,7,8}
C={8,9,10}
```

In [52]:
```
A|B
```

Out[52]:
```
{1, 2, 3, 4, 5, 6, 7, 8}
```

In [56]:
```
A.union(B)
A
```

Out[56]:
```
{1, 2, 3, 4, 5}
```

In [54]:
```
A.union(B,C)
```

Out[54]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In [57]:
```
"""updates the set calling the update() method with union of A,B&C.
For below example Set A will be updated with union of A,B & C."""
A.update(B,C)
A
```

Out[57]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

Intersection

In [58]:
```
A={1,2,3,4,5}
B={4,5,6,7,8}
```

In [59]:
```
A&B
```

Out[59]:
```
{4, 5}
```

In [60]:
```
A.intersection(B)
```

Out[60]:
```
{4, 5}
```

In [61]:
```
A.intersection(B)    Intersection of A and B
```

```
  Cell In[61], line 1
    A.intersection(B)    Intersection of A and B
                         ^
SyntaxError: invalid syntax
```

In [62]:
```
"""
Updates the set calling the intersection_update() method with the intersection
For below example Set A will be updated with the intersection of A & B.
"""
A.intersection_update(B)
A
```

```
Out[62]: {4, 5}
```

Difference

```
In [63]: A={1,2,3,4,5}
         B={4,5,6,7,8}
```

```
In [64]: A-B
```

```
Out[64]: {1, 2, 3}
```

```
In [65]: A.difference(B)
```

```
Out[65]: {1, 2, 3}
```

```
In [66]: B-A
```

```
Out[66]: {6, 7, 8}
```

```
In [67]: B.difference(A)
```

```
Out[67]: {6, 7, 8}
```

```
In [68]: '''
         Updates the set calling the differece_update() method with the difference of set
         For below example Set B will be updated with the difference fo B & A.
         '''
         B.difference_update(A)
         B
```

```
Out[68]: {6, 7, 8}
```

```
In [69]: A={1,2,3,4,5}
         B={4,5,6,7,8}
```

```
In [70]: A^B
```

```
Out[70]: {1, 2, 3, 6, 7, 8}
```

```
In [71]: A.symmetric_difference(B)
```

```
Out[71]: {1, 2, 3, 6, 7, 8}
```

```
In [72]: A
```

```
Out[72]: {1, 2, 3, 4, 5}
```

```
In [73]: A.symmetric_difference_update(B)
```

```
In [74]: A
```

```
Out[74]: {1, 2, 3, 6, 7, 8}
```

Subset,Superset & Disjoint

In [76]:
```
#superset=parent
#subset=child/kid
#Disjoint=neighbour
```

In [84]:
```
A={1,2,3,4,5,6,7,8,9}
B={3,4,5,6,7,8}
C={10,20,30,40}
```

In [85]:
```
B.issubset(A)
```

Out[85]:
```
True
```

In [86]:
```
A.issuperset(B)
```

Out[86]:
```
True
```

In [87]:
```
C.isdisjoint(A)
```

Out[87]:
```
True
```

In [88]:
```
B.isdisjoint(A)
```

Out[88]:
```
False
```

A

In [89]:
```
A
```

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

In [90]:
```
sum(A)
```

Out[90]:
```
45
```

In [91]:
```
max(A)
```

Out[91]:
```
9
```

In [92]:
```
min(A)
```

Out[92]:
```
1
```

In [93]:
```
len(A)
```

Out[93]:
```
9
```

In [94]:
```
list(enumerate(A))
```

Out[94]:
```
[(0, 1), (1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8), (8, 9)]
```

```
In [95]: D=sorted(A,reverse=True)
         D

Out[95]: [9, 8, 7, 6, 5, 4, 3, 2, 1]

In [96]: D

Out[96]: [9, 8, 7, 6, 5, 4, 3, 2, 1]

In [97]: sorted(D)

Out[97]: [1, 2, 3, 4, 5, 6, 7, 8, 9]

In [98]: E=sorted(C,reverse=True)
         E

Out[98]: [40, 30, 20, 10]

In [99]: sorted(E)

Out[99]: [10, 20, 30, 40]

In [ ]: 
```