

SET,DICTIONARY

```
In [2]: s={}
```

```
In [3]: s
```

```
Out[3]: {}
```

```
In [4]: s1=set()  
s1
```

```
Out[4]: set()
```

```
In [5]: s1={90,4,50,32,1}#set is ordered datatype but sometimes it is unordered also, so  
s1
```

```
Out[5]: {1, 4, 32, 50, 90}
```

```
In [6]: type(s1)
```

```
Out[6]: set
```

```
In [7]: s2={'z','o','a','d'}  
s2
```

```
Out[7]: {'a', 'd', 'o', 'z'}
```

```
In [8]: type(s2)
```

```
Out[8]: set
```

```
In [12]: print(s1)  
print(s2)
```

```
{32, 1, 50, 4, 90}  
{'a', 'z', 'd', 'o'}
```

```
In [11]: len(s2)
```

```
Out[11]: 4
```

```
In [13]: s3={1,2.3,"nit",(93+8j),False}  
s3
```

```
Out[13]: {(93+8j), 1, 2.3, False, 'nit'}
```

```
In [14]: s1.add(1)
```

```
In [15]: s1
```

```
Out[15]: {1, 4, 32, 50, 90}
```

```
In [16]: s1.add(100)
s1
```

```
Out[16]: {1, 4, 32, 50, 90, 100}
```

```
In [17]: s1.add(5)
s1
```

```
Out[17]: {1, 4, 5, 32, 50, 90, 100}
```

```
In [18]: s3.clear()
```

```
In [19]: s3
```

```
Out[19]: set()
```

```
In [20]: s1
```

```
Out[20]: {1, 4, 5, 32, 50, 90, 100}
```

```
In [21]: s4=s1.copy()
```

```
In [22]: s4
```

```
Out[22]: {1, 4, 5, 32, 50, 90, 100}
```

```
In [23]: s1[0] #index is not allowed in set
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[23], line 1
----> 1 s1[0]

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

```
In [24]: s1[1:5] #slicing is not allowed in set
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[24], line 1
----> 1 s1[1:5]

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

```
In [25]: s1
```

```
Out[25]: {1, 4, 5, 32, 50, 90, 100}
```

```
In [26]: s1.pop()
```

```
Out[26]: 32
```

```
In [27]: s1.pop() #Pop() eliminates random element
```

```
Out[27]: 1
```

```
In [28]: s1.pop(1) #pop() doesnot allowed indexes.
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[28], line 1  
----> 1 s1.pop(1)  
  
TypeError: set.pop() takes no arguments (1 given)
```

```
In [29]: s1.remove(4)
```

```
In [30]: s1
```

```
Out[30]: {5, 50, 90, 100}
```

```
In [35]: s1.remove(1000)
```

```
-----  
KeyError                                Traceback (most recent call last)  
Cell In[35], line 1  
----> 1 s1.remove(1000)  
  
KeyError: 1000
```

```
In [32]: s1.discard(1000)  
s1
```

```
Out[32]: {5, 50, 90, 100}
```

```
In [33]: 5 in s1
```

```
Out[33]: True
```

```
In [34]: 1000 in s1
```

```
Out[34]: False
```

set Operation

```
In [ ]: # union, intersect, difference, symmetric_difference, symmetric_difference_update
```

```
In [38]: a={1,2,3,4,5}  
b={4,5,6,7,8}  
c={8,9,10}
```

```
In [37]: a.union(b)
```

```
-----  
NameError                                Traceback (most recent call last)  
Cell In[37], line 1  
----> 1 a.union(b)  
  
NameError: name 'a' is not defined
```

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

```
c={8,9,10}
```

```
In [39]: a.union(b)
```

```
Out[39]: {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [40]: a.union(b,c)
```

```
Out[40]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [41]: a|b
```

```
Out[41]: {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [42]: a|b|c
```

```
Out[42]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [43]: print(a)
          print(b)
          print(c)
```

```
{1, 2, 3, 4, 5}
```

```
{4, 5, 6, 7, 8}
```

```
{8, 9, 10}
```

```
In [44]: a.intersection(b)
```

```
Out[44]: {4, 5}
```

```
In [45]: a.intersection(c)
```

```
Out[45]: set()
```

```
In [46]: a&b
```

```
Out[46]: {4, 5}
```

```
In [47]: a&c
```

```
Out[47]: set()
```

```
In [48]: print(a)
          print(b)
          print(c)
```

```
{1, 2, 3, 4, 5}
```

```
{4, 5, 6, 7, 8}
```

```
{8, 9, 10}
```

```
In [49]: a.difference(b)
```

```
Out[49]: {1, 2, 3}
```

```
In [50]: a.difference(c)
```

```
Out[50]: {1, 2, 3, 4, 5}
```

```
In [51]: a-b
```

```
Out[51]: {1, 2, 3}
```

```
In [52]: a-c
```

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

```
In [53]: print(a)
         print(b)
         print(c)
```

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

```
In [54]: b.difference_update(c)
```

```
In [55]: print(a)
         print(b)
         print(c)
```

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

```
In [56]: a.symmetric_difference(b)
```

```
Out[56]: {1, 2, 3, 6, 7}
```

```
In [57]: a^b
```

```
Out[57]: {1, 2, 3, 6, 7}
```

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

```
In [ ]:
```