

## set-Datastructure

```
In [2]: s={}
s
type(s)
```

Out[2]: dict

```
In [4]: s1=set() # empty set
print(s1)
type(s1)
```

set()

Out[4]: set

```
In [5]: s1.add(20)
s1
```

Out[5]: {20}

```
In [6]: s1.add(30,40) # we canot take more than 1 argument inside add() function.....
s1
```

```
-----
TypeError                                         Traceback (most recent call last)
Cell In[6], line 1
----> 1 s1.add(30,40)
      2 s1
```

`TypeError: set.add() takes exactly one argument (2 given)`

```
In [9]: s1.add(30)
s1.add(40)
s1.add(50)
print(s1)
```

{40, 50, 20, 30}

```
In [10]: s1
```

Out[10]: {20, 30, 40, 50}

```
In [16]: s1[0]
```

```
-----
TypeError                                         Traceback (most recent call last)
Cell In[16], line 1
----> 1 s1[0]
```

`TypeError: 'set' object is not subscriptable`

```
In [17]: s1
```

```
Out[17]: {10, 20, 30, 40, 50}
```

```
In [18]: s1[:]
```

```
-----  
TypeError
```

```
Traceback (most recent call last)
```

```
Cell In[18], line 1
```

```
----> 1 s1[:]
```

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

```
In [19]: s1
```

```
Out[19]: {10, 20, 30, 40, 50}
```

```
In [20]: s1.add([1,2,3])
```

```
s1
```

```
-----  
TypeError
```

```
Traceback (most recent call last)
```

```
Cell In[20], line 1
```

```
----> 1 s1.add([1,2,3])
```

```
2 s1
```

```
TypeError: unhashable type: 'list'
```

```
In [21]: s2=set()  
print(s2)
```

```
set()
```

```
In [24]: s2.add(1)
```

```
s2.add(2.3)
```

```
s2.add(True)
```

```
s2.add(1+3j)
```

```
s2.add('Abhi')
```

```
s2
```

```
Out[24]: {(1+3j), 1, 2.3, 'Abhi'}
```

```
In [25]: print(s)  
print(s1)  
print(s2)
```

```
{}
```

```
{40, 10, 50, 20, 30}
```

```
{1, 2.3, 'Abhi', (1+3j)}
```

```
In [26]: id(s) == id(s1)
```

```
Out[26]: False
```

```
In [27]: id(s1) == id(s2)
```

```
Out[27]: False
```

```
In [31]: s3=s2.copy()  
s3
```

```
Out[31]: {(1+3j), 1, 2.3, 'Abhi'}
```

```
In [33]: print(s)  
print(s1)  
print(s2)  
print(s3)
```

```
{  
}  
{40, 10, 50, 20, 30}  
{1, 2.3, 'Abhi', (1+3j)}  
{1, 2.3, 'Abhi', (1+3j)}
```

```
In [34]: s1
```

```
Out[34]: {10, 20, 30, 40, 50}
```

```
In [35]: s1.pop()
```

```
Out[35]: 40
```

```
In [36]: s1
```

```
Out[36]: {10, 20, 30, 50}
```

```
In [37]: s1.pop(3) # its takes no argument
```

```
-----  
TypeError  
Cell In[37], line 1  
----> 1 s1.pop(3)
```

```
Traceback (most recent call last)
```

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

```
In [38]: s1
```

```
Out[38]: {10, 20, 30, 50}
```

```
In [39]: s2
```

```
Out[39]: {(1+3j), 1, 2.3, 'Abhi'}
```

```
In [40]: s2.pop()
```

```
Out[40]: 1
```

```
In [41]: s2
```

```
Out[41]: {(1+3j), 2.3, 'Abhi'}
```

```
In [52]: s3
```

```
Out[52]: {(1+3j), 1, 2.3, 'Abhi'}
```

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

```
Out[55]: {2.3, 'Abhi'}
```

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

```
Out[56]: {2.3, 'Abhi'}
```

```
In [57]: print(s)
print(s1)
print(s2)
print(s3)
```

```
{}
{10, 50, 20, 30}
{2.3, 'Abhi', (1+3j)}
{2.3, 'Abhi'}
```

```
In [58]: s3.remove(1000) # it element is not available in the set it showing a Error.....
s3
```

```
-----
KeyError
Cell In[58], line 1
----> 1 s3.remove(1000)
      2 s3
```

Traceback (most recent call last)

```
KeyError: 1000
```

```
In [60]: s3
```

```
Out[60]: {2.3, 'Abhi'}
```

```
In [61]: s3.discard(1000)
s3
```

```
Out[61]: {2.3, 'Abhi'}
```

```
In [63]: s3.discard(True)
s3
```

```
Out[63]: {2.3, 'Abhi'}
```

```
In [65]: for i in s1:
    print(i)
```

```
10
50
20
30
```

s1

```
In [67]: for i in enumerate(s1):print(i)
```

```
(0, 10)
(1, 50)
(2, 20)
(3, 30)
```

```
In [68]: s3
```

```
Out[68]: {2.3, 'Abhi'}
```

```
In [76]: s5={10, 'True', 1+3j, 'nit', 2.3}
s5
```

```
Out[76]: {(1+3j), 10, 2.3, 'True', 'nit'}
```

```
In [77]: len(s5)
```

```
Out[77]: 5
```

```
In [78]: print(s5)
```

```
{2.3, 10, 'True', (1+3j), 'nit'}
```

```
In [80]: s5.discard(200) #
s5
```

```
Out[80]: {(1+3j), 10, 2.3, 'True', 'nit'}
```

```
In [81]: s5.remove(200) # here it element is not available in the list
s5
```

-----  
**KeyError**  
Cell In[81], line 1  
----> 1 s5.remove(200)  
2 s5

Traceback (most recent call last)

**KeyError:** 200

```
In [82]: s5
```

```
Out[82]: {(1+3j), 10, 2.3, 'True', 'nit'}
```

```
In [83]: s5.remove((1+3j))
s5
```

```
Out[83]: {10, 2.3, 'True', 'nit'}
```

```
In [84]: s5.add((1+3j)) # we know at a time multiple argument we canot take here
s5
```

```
Out[84]: {(1+3j), 10, 2.3, 'True', 'nit'}
```

```
In [ ]: s5.
```

```
In [85]: s5.pop()  
s5
```

```
Out[85]: {(1+3j), 10, 'True', 'nit'}
```

```
In [86]: s6=s5.copy()  
s6
```

```
Out[86]: {(1+3j), 10, 'True', 'nit'}
```

```
In [91]: s5.add('abhi')  
s5
```

```
Out[91]: {(1+3j), 10, 'True', 'abhi', 'nit', 'ram'}
```

```
In [92]: print(s5)  
print(s6)
```

```
{10, 'True', 'abhi', 'ram', (1+3j), 'nit'}  
{'True', (1+3j), 10, 'nit', 'abhi'}
```

```
In [93]: print(s)  
print(s1)  
print(s2)  
print(s3)  
print(s4)  
print(s5)  
print(s6)
```

```
{}  
{10, 50, 20, 30}  
{2.3, 'Abhi', (1+3j)}  
{'Abhi'}  
{'Abhi'}  
{10, 'True', 'abhi', 'ram', (1+3j), 'nit'}  
{'True', (1+3j), 10, 'nit', 'abhi'}
```

```
In [94]: for i in s5:  
    print(i)
```

```
10  
True  
abhi  
ram  
(1+3j)  
nit
```

```
In [100...]: for i in enumerate(s5):  
    print(i)
```

```
(0, 10)
(1, 'True')
(2, 'abhi')
(3, 'ram')
(4, (1+3j))
(5, 'nit')
```

```
In [101]: s5
```

```
Out[101]: {(1+3j), 10, 'True', 'abhi', 'nit', 'ram'}
```

```
In [102]: s5.remove((10))
s5
```

```
Out[102]: {(1+3j), 'True', 'abhi', 'nit', 'ram'}
```

```
In [103]: s5.remove('ram')
s5
```

```
Out[103]: {(1+3j), 'True', 'abhi', 'nit'}
```

```
In [104]: s5.discard(1+3j)
s5
```

```
Out[104]: {'True', 'abhi', 'nit'}
```

```
In [2]: s7={1,2,5,7,10}
s7
```

```
Out[2]: {1, 2, 5, 7, 10}
```

```
In [3]: s7.add(3)
s7
```

```
Out[3]: {1, 2, 3, 5, 7, 10}
```

```
In [4]: s7.add('nit')
s7
```

```
Out[4]: {1, 10, 2, 3, 5, 7, 'nit'}
```

```
In [5]: s7
```

```
Out[5]: {1, 10, 2, 3, 5, 7, 'nit'}
```

```
In [6]: s7.clear()
s7
```

```
Out[6]: set()
```

```
In [7]: s7
```

```
Out[7]: set()
```

# SET-Opearation

## Union (|)

```
In [13]: s1={1,2,3,4,5}  
s2={4,5,6,7,8}  
s3={8,9,10}
```

```
In [14]: s1 | s2 #union
```

```
Out[14]: {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [16]: s1 | s2|s3
```

```
Out[16]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [17]: s2 | s3
```

```
Out[17]: {4, 5, 6, 7, 8, 9, 10}
```

```
In [18]: s3 | s1
```

```
Out[18]: {1, 2, 3, 4, 5, 8, 9, 10}
```

## Intersection (&)

```
In [19]: s1.intersection(s2)
```

```
Out[19]: {4, 5}
```

```
In [20]: s1 & s2
```

```
Out[20]: {4, 5}
```

```
In [21]: s2.intersection(s3)
```

```
Out[21]: {8}
```

```
In [22]: s2 & s3
```

```
Out[22]: {8}
```

```
In [23]: s1.intersection(s3)
```

```
Out[23]: set()
```

```
In [24]: s2 & s1
```

```
Out[24]: {4, 5}
```

```
In [25]: s1={1,2,3,4,5}  
s2={4,5,6,7,8}  
s3={8,9,10}
```

## Difference (-)

```
In [26]: s1.difference(s2)
```

```
Out[26]: {1, 2, 3}
```

```
In [27]: s2.difference(s3)
```

```
Out[27]: {4, 5, 6, 7}
```

```
In [29]: s1.difference(s3)
```

```
Out[29]: {1, 2, 3, 4, 5}
```

```
In [31]: s3.difference(s1)
```

```
Out[31]: {8, 9, 10}
```

## Symmetric\_difference (^)

```
In [32]: s1={1,2,3,4,5}  
s2={4,5,6,7,8}  
s3={8,9,10}
```

```
In [34]: s1.symmetric_difference(s2)
```

```
Out[34]: {1, 2, 3, 6, 7, 8}
```

```
In [35]: s2.symmetric_difference(s3)
```

```
Out[35]: {4, 5, 6, 7, 9, 10}
```

```
In [36]: s3.symmetric_difference(s1)
```

```
Out[36]: {1, 2, 3, 4, 5, 8, 9, 10}
```

```
In [37]: s1.symmetric_difference(s3)
```

```
Out[37]: {1, 2, 3, 4, 5, 8, 9, 10}
```

```
In [38]: s2.symmetric_difference(s1)
```

```
Out[38]: {1, 2, 3, 6, 7, 8}
```

## symmetric\_difference\_update (^)

```
In [39]: s1={1,2,3,4,5} # here common part will not be included
s2={4,5,6,7,8}
s3={8,9,10}
```

```
In [42]: s1.symmetric_difference_update(s2)
s1
```

```
Out[42]: {1, 2, 3, 6, 7, 8}
```

```
In [43]: s2.symmetric_difference_update(s3)
s2
```

```
Out[43]: {4, 5, 6, 7, 9, 10}
```

```
In [44]: s3.symmetric_difference_update(s1)
s3
```

```
Out[44]: {1, 2, 3, 6, 7, 9, 10}
```

```
In [46]: s1 ^(s2)
s1
```

```
Out[46]: {1, 2, 3, 6, 7, 8}
```

```
In [2]: s1={1,2,3,4,5}
s2={4,5,6,7,8}
s3={8,9,10}
```

```
In [3]: s1
```

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

```
In [4]: s1.add(10)
s1
```

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

```
In [27]: s1.add(6)
s1
```

```
Out[27]: {1, 2, 3, 4, 5, 6, 10}
```

# Superset (Super class)(dad) ,Subset(son), disjoint(neighbour)

```
In [3]: ss1={1,2,3,4,5,6,7,8,9}
ss2={3,4,5,6,7,8}
ss3={10,20,30,40}
```

```
In [4]: ss1.issuperset(ss2)
```

Out[4]: True

```
In [5]: ss2.issubset(ss1)
```

Out[5]: True

```
In [6]: print(ss1)
print(ss2)
print(ss3)
```

```
{1, 2, 3, 4, 5, 6, 7, 8, 9}
{3, 4, 5, 6, 7, 8}
{40, 10, 20, 30}
```

```
In [7]: ss3.isdisjoint(ss2)
```

Out[7]: True

```
In [8]: ss3.isdisjoint(ss1)
```

Out[8]: True

```
In [9]: ss4={1,2,3,4,5}
ss5={6,7,8}
ss6={9,10}
```

```
In [10]: ss4.isdisjoint(ss5)
```

Out[10]: True

```
In [11]: ss4.issuperset(ss5)
```

Out[11]: False

```
In [12]: ss4.issubset(ss5)
```

Out[12]: False

## DICTIONARY key-values

```
In [15]: d={}# empty dict  
d
```

```
Out[15]: {}
```

```
In [16]: type(d)
```

```
Out[16]: dict
```

```
In [17]: set()
```

```
Out[17]: set()
```

```
In [18]: list()
```

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

```
In [19]: tuple()
```

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

```
In [20]: mydict={1:'one',2:'two',3:'three'}  
mydict
```

```
Out[20]: {1: 'one', 2: 'two', 3: 'three'}
```

```
In [22]: mydict.keys()
```

```
Out[22]: dict_keys([1, 2, 3])
```

```
In [23]: mydict.values()
```

```
Out[23]: dict_values(['one', 'two', 'three'])
```

```
In [25]: mydict.items()
```

```
Out[25]: dict_items([(1, 'one'), (2, 'two'), (3, 'three')])
```

```
In [27]: mydict[4]='four'  
mydict
```

```
Out[27]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [28]: mydict[5]='five'  
mydict
```

```
Out[28]: {1: 'one', 2: 'two', 3: 'three', 4: 'four', 5: 'five'}
```

```
In [30]: mydict[1+2j]='complex'  
mydict[True]='boolean'  
mydict[1.2]='float'  
mydict[10]='intiger'
```

```
mydict['abhi']='string'
mydict
```

```
Out[30]: {1: 'boolean',
2: 'two',
3: 'three',
4: 'four',
5: 'five',
(1+2j): 'complex',
1.2: 'float',
10: 'intiger',
'abhi': 'string'}
```

```
In [36]: keys={'a','b','c','d'}
mydict3=dict.fromkeys(keys)
mydict3
```

```
Out[36]: {'d': None, 'a': None, 'c': None, 'b': None}
```

```
In [37]: print(type(keys))
print(type(mydict3))
```

```
<class 'set'>
<class 'dict'>
```

```
In [35]: keys={'a','b','c','d'}
value=10
mydict3=dict.fromkeys(keys,value)
mydict3
```

```
Out[35]: {'d': 10, 'a': 10, 'c': 10, 'b': 10}
```

```
In [39]: keys={'a','b','c','d'}
value=[10,20,30]
mydict3=dict.fromkeys(keys,value)
mydict3
```

```
Out[39]: {'d': [10, 20, 30], 'a': [10, 20, 30], 'c': [10, 20, 30], 'b': [10, 20, 30]}
```

```
In [42]: value.append(40)
mydict3
```

```
Out[42]: {'d': [10, 20, 30, 40, 40],
'a': [10, 20, 30, 40, 40],
'c': [10, 20, 30, 40, 40],
'b': [10, 20, 30, 40, 40]}
```

```
In [43]: mydict
```

```
Out[43]: {1: 'boolean',
 2: 'two',
 3: 'three',
 4: 'four',
 5: 'five',
 (1+2j): 'complex',
 1.2: 'float',
 10: 'intiger',
 'abhi': 'string'}
```

```
In [45]: mydict[1]
```

```
Out[45]: 'boolean'
```

```
In [50]: mydict[2]
```

```
Out[50]: 'two'
```

```
In [51]: mydict[4]
```

```
Out[51]: 'four'
```

```
In [52]: mydict[10]
```

```
Out[52]: 'intiger'
```

## Range

```
In [53]: range()
```

-----  
**TypeError**  
 Cell In[53], line 1  
 ----> 1 range()

Traceback (most recent call last)

**TypeError**: range expected at least 1 argument, got 0

```
In [54]: range(10)
```

```
Out[54]: range(0, 10)
```

```
In [1]: s1=set()  
print(s1)
```

set()

```
In [4]: s1={1,2,3,4,5}  
print(s1)
```

{1, 2, 3, 4, 5}

```
In [5]: s2={3,4,5,6,10}  
s2
```

```
Out[5]: {3, 4, 5, 6, 10}
```

```
In [6]: print(s1)
print(s2)
```

```
{1, 2, 3, 4, 5}
{3, 4, 5, 6, 10}
```

```
In [16]: s1^s2
s1
```

```
Out[16]: {1, 2, 6, 10}
```

```
In [8]: type(s1)
```

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

```
In [18]: print(s1)
print(s2)
```

```
{1, 2, 6, 10}
{3, 4, 5, 6, 10}
```

```
In [19]: s1.issuperset(s2)
```

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

```
In [20]: s1.clear()
s1
```

```
Out[20]: set()
```

```
In [22]: s1={3,4,5,6,7}
s1
```

```
Out[22]: {3, 4, 5, 6, 7}
```

```
In [23]: print(s1)
print(s2)
```

```
{3, 4, 5, 6, 7}
{3, 4, 5, 6, 10}
```

```
In [24]: s1.issuperset(s2)
```

```
Out[24]: False
```

```
In [25]: s1.pop()
s1
```

```
Out[25]: {4, 5, 6, 7}
```

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

```
Out[26]: set()
```

```
In [27]: s1={3,4,5,6}  
s1
```

```
Out[27]: {3, 4, 5, 6}
```

```
In [28]: print(s1)  
print(s2)
```

```
{3, 4, 5, 6}  
{3, 4, 5, 6, 10}
```

```
In [3]: s1={3,4,5,6}  
print(s1)
```

```
{3, 4, 5, 6}
```

```
In [6]: s2={3,4,5,6,7,8,9}  
print(s2)
```

```
{3, 4, 5, 6, 7, 8, 9}
```

```
In [7]: print(s1)  
print(s2)
```

```
{3, 4, 5, 6}  
{3, 4, 5, 6, 7, 8, 9}
```

```
In [8]: s1.issuperset(s2)  
s1
```

```
Out[8]: {3, 4, 5, 6}
```

```
In [8]: ss4={1,2,3,4,5}  
ss5={6,7,8}  
ss6={9,10}
```

```
In [9]: ss5.isdisjoint(ss4)
```

```
Out[9]: True
```

```
In [3]: mydict={}  
mydict
```

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

```
In [4]: type(mydict)
```

```
Out[4]: dict
```

```
In [6]: mydict={1:'one',2:'two',3:'three',4:'four'}  
mydict
```

```
Out[6]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [8]: mydict.keys()
```

```
Out[8]: dict_keys([1, 2, 3, 4])
```

```
In [9]: mydict.values()
```

```
Out[9]: dict_values(['one', 'two', 'three', 'four'])
```

```
In [10]: mydict.items()
```

```
Out[10]: dict_items([(1, 'one'), (2, 'two'), (3, 'three'), (4, 'four')])
```

```
In [11]: mydict
```

```
Out[11]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [13]: mydict[5]='five'
```

```
mydict
```

```
Out[13]: {1: 'one', 2: 'two', 3: 'three', 4: 'four', 5: 'five'}
```

```
In [16]: keys={'a','b','c','d'}
```

```
mydict3=dict.fromkeys(keys)
```

```
mydict3
```

```
Out[16]: {'a': None, 'c': None, 'd': None, 'b': None}
```

```
In [18]: print(type(keys))
```

```
print(type(mydict3))
```

```
<class 'set'>
```

```
<class 'dict'>
```

```
In [19]: keys={'a','b','c','d'}
```

```
value=20
```

```
mydict3=dict.fromkeys(keys,value)
```

```
mydict3
```

```
Out[19]: {'a': 20, 'c': 20, 'd': 20, 'b': 20}
```

```
In [33]: mydict.keys()
```

```
Out[33]: dict_keys([1, 2, 3, 4, 5])
```

```
In [36]: mydict={'A':10,'B':20,'C':30,'d':40}
```

```
mydict
```

```
Out[36]: {'A': 10, 'B': 20, 'C': 30, 'd': 40}
```

```
In [37]: mydict.keys()
```

```
Out[37]: dict_keys(['A', 'B', 'C', 'd'])
```

```
In [38]: mydict.values()
```

```
Out[38]: dict_values([10, 20, 30, 40])
```

```
In [39]: mydict.items()
```

```
Out[39]: dict_items([('A', 10), ('B', 20), ('C', 30), ('D', 40)])
```

```
In [41]: print(type(keys))
print(type(values))
```

```
<class 'set'>
<class 'list'>
```

```
In [49]: keys={'A','B','C','D'}
values={100}
mydict=dict.fromkeys(keys,values)
mydict
```

```
Out[49]: {'C': {100}, 'A': {100}, 'B': {100}, 'D': {100}}
```

```
In [50]: print(type(keys))
print(type(values))
```

```
<class 'set'>
<class 'set'>
```

```
In [51]: mydict
```

```
Out[51]: {'C': {100}, 'A': {100}, 'B': {100}, 'D': {100}}
```

```
In [53]: type(mydict)
```

```
Out[53]: dict
```

```
In [62]: mydict.clear()
mydict
```

```
Out[62]: {}
```

```
In [64]: mydict={1:'one',2:'two',3:'three',4:'four'}# dictionary with integer keys
mydict
```

```
Out[64]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [67]: mydict=dict({1:'one',2:'two',3:'three',4:'four'}) # create dictionary using dict ()
mydict
```

```
Out[67]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [2]: mydict=dict({1:'one',2:'two',3:'three',4:'four'}) # create dictionary using dict ()
mydict
```

```
Out[2]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [6]: mydict.keys()
```

```
Out[6]: dict_keys([1, 2, 3, 4])
```

```
In [7]: mydict.values()
```

```
Out[7]: dict_values(['one', 'two', 'three', 'four'])
```

```
In [8]: mydict.items()
```

```
Out[8]: dict_items([(1, 'one'), (2, 'two'), (3, 'three'), (4, 'four')])
```

```
In [30]: mydict.keys()
```

```
Out[30]: dict_keys([1, 2, 3, 4])
```

```
In [21]: key={1,2,3,4}
values=['one','two','three','four']
mydict=dict.fromkeys(key,values)
mydict
```

```
Out[21]: {1: ['one', 'two', 'three', 'four'],
2: ['one', 'two', 'three', 'four'],
3: ['one', 'two', 'three', 'four'],
4: ['one', 'two', 'three', 'four']}
```

```
In [23]: print(type(key))
print(type(values))
```

```
<class 'set'>
<class 'list'>
```

```
In [29]: values.append('five')
values
```

```
Out[29]: ['one', 'two', 'three', 'four', 'five', 'five', 'five']
```

```
In [32]: mydict={1:'one',2:'two','A':['aryan','john','maria']} #dictionary with
mydict
```

```
Out[32]: {1: 'one', 2: 'two', 'A': ['aryan', 'john', 'maria']}
```

```
In [34]: mydict={1:'one',2:'two','A':['arif','john','ram'],'B':('Bat','cat')}
mydict
```

```
Out[34]: {1: 'one', 2: 'two', 'A': ['arif', 'john', 'ram'], 'B': ('Bat', 'cat')}
```

```
In [35]: mydict.keys()
```

```
Out[35]: dict_keys([1, 2, 'A', 'B'])
```

```
In [36]: mydict.values()
```

```
Out[36]: dict_values(['one', 'two', ['arif', 'john', 'ram'], ('Bat', 'cat')])
```

```
In [38]: mydict.items()
```

```
Out[38]: dict_items([(1, 'one'), (2, 'two'), ('A', ['arif', 'john', 'ram']), ('B', ('Bat', 'cat'))])
```

```
In [17]: keys={1,2,'A','B'}
values=['lakshman'],('boss')
mydict=dict.fromkeys(keys,values)
mydict
```

```
Out[17]: {'A': (['lakshman'], 'boss'),
 1: (['lakshman'], 'boss'),
 2: (['lakshman'], 'boss'),
 'B': (['lakshman'], 'boss')}
```

```
In [41]: 
```

```
Out[41]: {'A': (['lakshman'], 'bos'),
 1: (['lakshman'], 'bos'),
 2: (['lakshman'], 'bos'),
 'B': (['lakshman'], 'bos')}
```

```
In [43]: keys={'a','b','c','d'}
mydict=dict.fromkeys(keys)
mydict
```

```
Out[43]: {'c': None, 'a': None, 'b': None, 'd': None}
```

```
In [44]: keys={'a','b','c','d'}
values=10
mydict=dict.fromkeys(keys,values)
mydict
```

```
Out[44]: {'c': 10, 'a': 10, 'b': 10, 'd': 10}
```

```
In [46]: keys={'a','b','c','d'}
values=[10,20,30]
mydict=dict.fromkeys(keys,values)
mydict
```

```
Out[46]: {'c': [10, 20, 30], 'a': [10, 20, 30], 'b': [10, 20, 30], 'd': [10, 20, 30]}
```

```
In [47]: print(type(keys))
print(type(values))
```

```
<class 'set'>
<class 'list'>
```

```
In [53]: values.append(50)
mydict
```

```
Out[53]: {'c': [10, 20, 30, 40, 40, 40, 50],
           'a': [10, 20, 30, 40, 40, 40, 50],
           'b': [10, 20, 30, 40, 40, 40, 50],
           'd': [10, 20, 30, 40, 40, 40, 50]}
```

## Accessing Items

```
In [2]: mydict={1:'one',2:'two',3:'three',4:'four'}
mydict
```

```
Out[2]: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [3]: mydict[1] # Access item using key
```

```
Out[3]: 'one'
```

```
In [4]: mydict[4]
```

```
Out[4]: 'four'
```

```
In [5]: mydict.get(1) #Access item using get() method
```

```
Out[5]: 'one'
```

```
In [6]: mydict.get(3)
```

```
Out[6]: 'three'
```

```
In [7]: mydict1={'Name':'Abhi','Age':23,'id':1010,'DOB':2004,'job':'Reacher'}
mydict1
```

```
Out[7]: {'Name': 'Abhi', 'Age': 23, 'id': 1010, 'DOB': 2004, 'job': 'Reacher'}
```

```
In [8]: mydict1['Name'] # Access item using key
```

```
Out[8]: 'Abhi'
```

```
In [12]: mydict1['Age']
```

```
Out[12]: 23
```

```
In [14]: mydict1['job']
```

```
Out[14]: 'Reacher'
```

```
In [15]: mydict1['DOB']
```

```
Out[15]: 2004
```

```
In [16]: mydict2={'id1':'john','id2':'Decruz','id3':'shyam','id4':'Aliva','id5':'Happy'}
```

```
mydict2  
Out[16]: {'id1': 'john',  
          'id2': 'Decruz',  
          'id3': 'shyam',  
          'id4': 'Aliva',  
          'id5': 'Happy'}
```

```
In [18]: mydict2['id1']
```

```
Out[18]: 'john'
```

```
In [19]: mydict2['id2']
```

```
Out[19]: 'Decruz'
```

```
In [21]: mydict2['id3']
```

```
Out[21]: 'shyam'
```

```
In [22]: mydict2['id4']
```

```
Out[22]: 'Aliva'
```

```
In [23]: mydict2.get('id3')
```

```
Out[23]: 'shyam'
```

```
In [25]: mydict2.get('id4')
```

```
Out[25]: 'Aliva'
```

```
In [29]: mydict2.get('Aliva')  
mydict2
```

```
Out[29]: {'id1': 'john',  
          'id2': 'Decruz',  
          'id3': 'shyam',  
          'id4': 'Aliva',  
          'id5': 'Happy'}
```

## Add, Remove & Change items

```
In [30]: mydict2
```

```
Out[30]: {'id1': 'john',  
          'id2': 'Decruz',  
          'id3': 'shyam',  
          'id4': 'Aliva',  
          'id5': 'Happy'}
```

```
In [32]: mydict2['id1'] # change Dictionary items
mydict2['id2']
```

Out[32]: 'Decruz'

```
In [33]: mydict2
```

```
Out[33]: {'id1': 'john',
'id2': 'Decruz',
'id3': 'shyam',
'id4': 'Aliva',
'id5': 'Happy'}
```

```
In [35]: dict2={'id1':'Abhishek'}
mydict2.update(dict2)
mydict2
```

```
Out[35]: {'id1': 'Abhishek',
'id2': 'Decruz',
'id3': 'shyam',
'id4': 'Aliva',
'id5': 'Happy'}
```

```
In [36]: mydict3={'Book':100,'pen':200,'pencil':25,'shapner':10,'Eraser':10,}
mydict3
```

Out[36]: {'Book': 100, 'pen': 200, 'pencil': 25, 'shapner': 10, 'Eraser': 10}

```
In [37]: len(mydict3)
```

Out[37]: 5

```
In [46]: mydict4={'Book':100,'pen':200,'pencil':25,'shapner':10,'Eraser':10,}
mydict4
```

Out[46]: {'Book': 100, 'pen': 200, 'pencil': 25, 'shapner': 10, 'Eraser': 10}

```
In [47]: dict4={'Book':150}
mydict4.update(dict4)
mydict4
```

Out[47]: {'Book': 150, 'pen': 200, 'pencil': 25, 'shapner': 10, 'Eraser': 10}

```
In [51]: dict4={'pen':250}
mydict4.update(dict4)
mydict4
```

Out[51]: {'Book': 150, 'pen': 250, 'pencil': 25, 'shapner': 10, 'Eraser': 10}

```
In [52]: mydict4
```

Out[52]: {'Book': 150, 'pen': 250, 'pencil': 25, 'shapner': 10, 'Eraser': 10}

```
In [53]: mydict4['pencil']='Natraj' # item in the Dictionary
mydict4
```

```
Out[53]: {'Book': 150, 'pen': 250, 'pencil': 'Natraj', 'shapner': 10, 'Eraser': 10}
```

```
In [55]: mydict4['Book']='Attitude life'
mydict4
```

```
Out[55]: {'Book': 'Attitude life',
          'pen': 250,
          'pencil': 'Natraj',
          'shapner': 10,
          'Eraser': 10}
```

```
In [56]: dict4={'shapner':20.04}
mydict4.update(dict4)
mydict4
```

```
Out[56]: {'Book': 'Attitude life',
          'pen': 250,
          'pencil': 'Natraj',
          'shapner': 20.04,
          'Eraser': 10}
```

```
In [57]: mydict4.pop('Eraser') # Removing items in the dictionary using pop method
mydict4
```

```
Out[57]: {'Book': 'Attitude life', 'pen': 250, 'pencil': 'Natraj', 'shapner': 20.04}
```

```
In [58]: mydict4
```

```
Out[58]: {'Book': 'Attitude life', 'pen': 250, 'pencil': 'Natraj', 'shapner': 20.04}
```

```
In [59]: mydict4.popitem() # A random item is removed
```

```
Out[59]: ('shapner', 20.04)
```

```
In [60]: mydict4
```

```
Out[60]: {'Book': 'Attitude life', 'pen': 250, 'pencil': 'Natraj'}
```

```
In [62]: mydict4.pop('pencil')
```

```
Out[62]: 'Natraj'
```

```
In [63]: mydict4
```

```
Out[63]: {'Book': 'Attitude life', 'pen': 250}
```

```
In [64]: del[mydict4['Book']]
mydict4
```

```
Out[64]: {'pen': 250}
```

```
In [65]: mydict4
```

```
Out[65]: {'pen': 250}
```

```
In [66]: mydict4['shapner']=100 # Add items
mydict4
```

```
Out[66]: {'pen': 250, 'shapner': 100}
```

```
In [68]: dict4={'shapner':200} # Update items in the dict
mydict4.update(dict4)
mydict4
```

```
Out[68]: {'pen': 250, 'shapner': 100, 'shapner': 200}
```

```
In [70]: mydict4['Eraser']=10
mydict4
```

```
Out[70]: {'pen': 250, 'shapner': 100, 'shapner': 200, 'Eraser': 10}
```

```
In [2]: mydict5={'Name':'abhishek','id':2010,'Age':24,'state':'odisha','District':'puri'}
mydict5
```

```
Out[2]: {'Name': 'abhishek',
         'id': 2010,
         'Age': 24,
         'state': 'odisha',
         'District': 'puri'}
```

```
In [4]: mydict5['Name']='Arif' # change the name
mydict5
```

```
Out[4]: {'Name': 'Arif', 'id': 2010, 'Age': 24, 'state': 'odisha', 'District': 'puri'}
```

```
In [5]: dict5={'Name':'Atif'} # update the name
mydict5.update(dict5)
mydict5
```

```
Out[5]: {'Name': 'Atif', 'id': 2010, 'Age': 24, 'state': 'odisha', 'District': 'puri'}
```

```
In [6]: mydict5['job']='Data Architect' # Add item in the Dictionary
mydict5
```

```
Out[6]: {'Name': 'Atif',
         'id': 2010,
         'Age': 24,
         'state': 'odisha',
         'District': 'puri',
         'job': 'Data Architect'}
```

```
In [11]: mydict5['job']='Data Atrchitect'
mydict5
```

```
Out[11]: {'Name': 'Atif',
          'id': 2010,
          'Age': 24,
          'state': 'odisha',
          'District': 'puri',
          'job': 'Data Architect'}
```

```
In [12]: mydict5
```

```
Out[12]: {'Name': 'Atif',
          'id': 2010,
          'Age': 24,
          'state': 'odisha',
          'District': 'puri',
          'job': 'Data Architect'}
```

```
In [13]: mydict5.pop('job')# Removing item in the dictionary usong pop method
mydict5
```

```
Out[13]: {'Name': 'Atif', 'id': 2010, 'Age': 24, 'state': 'odisha', 'District': 'puri'}
```

```
In [14]: mydict5
```

```
Out[14]: {'Name': 'Atif', 'id': 2010, 'Age': 24, 'state': 'odisha', 'District': 'puri'}
```

```
In [15]: mydict5.popitem()# A random item is removed
mydict5
```

```
Out[15]: {'Name': 'Atif', 'id': 2010, 'Age': 24, 'state': 'odisha'}
```

```
In [16]: del[mydict5['Name']]# Removing item using del method
mydict5
```

```
Out[16]: {'id': 2010, 'Age': 24, 'state': 'odisha'}
```

```
In [17]: mydict5.clear() # Delete all items of the dictionary using clear method
mydict5
```

```
Out[17]: {}
```

```
In [1]: mydict5={'Name':'abhishek','id':2010,'Age':24,'state':'odisha','District':'puri'}
mydict5
```

```
Out[1]: {'Name': 'abhishek',
          'id': 2010,
          'Age': 24,
          'state': 'odisha',
          'District': 'puri'}
```

```
In [2]: del[mydict5['Age']]
mydict5
```

```
Out[2]: {'Name': 'abhishek', 'id': 2010, 'state': 'odisha', 'District': 'puri'}
```

```
In [3]: mydict5['Age']=23
mydict5
```

```
Out[3]: {'Name': 'abhishek',
'id': 2010,
'state': 'odisha',
'District': 'puri',
'Age': 23}
```

```
In [5]: dict5={'Age':30}
mydict5.update(dict5)
mydict5
```

```
Out[5]: {'Name': 'abhishek',
'id': 2010,
'state': 'odisha',
'District': 'puri',
'Age': 30}
```

```
In [6]: mydict5['Age']=40
mydict5
```

```
Out[6]: {'Name': 'abhishek',
'id': 2010,
'state': 'odisha',
'District': 'puri',
'Age': 40}
```

```
In [7]: mydict5.pop('Age') # Remove item in the dictionary using pop method
mydict5
```

```
Out[7]: {'Name': 'abhishek', 'id': 2010, 'state': 'odisha', 'District': 'puri'}
```

```
In [8]: mydict5.popitem() # Remove random item in the dictionary
mydict5
```

```
Out[8]: {'Name': 'abhishek', 'id': 2010, 'state': 'odisha'}
```

## Copy Dictionary

```
In [1]: mydict={'Name':'Asif','ID':1234,'DOB':1991,'Address':'Odisha'}
mydict
```

```
Out[1]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [2]: mydict1=mydict# create a new reference
mydict1
```

```
Out[2]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [3]: print(mydict)
print(mydict1)
```

```
{'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
{'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [4]: id(mydict) == id(mydict1)
```

```
Out[4]: True
```

```
In [5]: id(mydict),id(mydict1) #here those variable value id are same and address are same
```

```
Out[5]: (2644598072768, 2644598072768)
```

```
In [7]: mydict2=mydict.copy()
mydict
```

```
Out[7]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [8]: print(mydict)
print(mydict1)
print(mydict2)
```

```
{'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
{'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
{'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [9]: id(mydict),id(mydict1),id(mydict2)
```

```
Out[9]: (2644598072768, 2644598072768, 2644598097792)
```

```
In [10]: id(mydict2)
```

```
Out[10]: 2644598097792
```

```
In [11]: mydict['Address']='Mumbai'
mydict
```

```
Out[11]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Mumbai'}
```

```
In [13]: dict={'Address':'Delhi'}
mydict.update(dict)
mydict
```

```
Out[13]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Delhi'}
```

```
In [14]: mydict1
```

```
Out[14]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Delhi'}
```

```
In [15]: mydict2
```

```
Out[15]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

# Loop through a Dictionary

```
In [16]: mydict1
```

```
Out[16]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Delhi'}
```

```
In [17]: for i in mydict1:  
    print(i)# Dictionary items
```

Name  
ID  
DOB  
Address

```
In [18]: for i in enumerate(mydict1):  
    print(i)
```

(0, 'Name')  
(1, 'ID')  
(2, 'DOB')  
(3, 'Address')

```
In [21]: for i in mydict1:  
    print(i,'->',mydict[i]) #key value and pair
```

Name -> Asif  
ID -> 1234  
DOB -> 1991  
Address -> Delhi

```
In [24]: for i in mydict1:  
    print(i,'-',mydict[i])
```

Name -> Asif  
ID -> 1234  
DOB -> 1991  
Address -> Delhi

# Dictionary Membership

```
In [26]: mydict={'Name':'Asif','ID':1234,'DOB':1991,'Address':'Odisha'}  
mydict
```

```
Out[26]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [27]: 'Name' in mydict
```

```
Out[27]: True
```

```
In [28]: 'Title' in mydict
```

```
Out[28]: False
```

```
In [29]: 'pincode' in mydict
```

```
Out[29]: False
```

```
In [3]: mydict1={'Name':'Asif', 'ID':1234, 'DOB':1991, 'Address':'Odisha'}
mydict1
```

```
Out[3]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [4]: 'Name' in mydict1
```

```
Out[4]: True
```

## All/Any

```
In [5]: mydict={'Name':'Asif', 'ID':1234, 'DOB':1991, 'Address':'Odisha'}
mydict
```

```
Out[5]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [6]: all(mydict)
```

```
Out[6]: True
```

```
In [7]: any(mydict)
```

```
Out[7]: True
```

```
In [8]: mydict['value']=0
mydict
```

```
Out[8]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha', 'value': 0}
```

```
In [9]: all(mydict)
```

```
Out[9]: True
```

```
In [10]: any(mydict)
```

```
Out[10]: True
```

```
In [15]: dict={'value':0.111}
mydict.update(dict)
mydict
```

```
Out[15]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha', 'value': 0.111}
```

```
In [16]: mydict.pop('value')
mydict
```

```
Out[16]: {'Name': 'Asif', 'ID': 1234, 'DOB': 1991, 'Address': 'Odisha'}
```

```
In [ ]:
```

## range

```
In [1]: range()
```

```
-----  
TypeError                                     Traceback (most recent call last)  
Cell In[1], line 1  
----> 1 range()  
TypeError: range expected at least 1 argument, got 0
```

```
In [2]: range(5) #0to 5
```

```
Out[2]: range(0, 5)
```

```
In [3]: list(range(0,5))
```

```
Out[3]: [0, 1, 2, 3, 4]
```

```
In [4]: range(10,20)
```

```
Out[4]: range(10, 20)
```

```
In [5]: list(range(10,20))
```

```
Out[5]: [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

```
In [6]: range(0,100,10)
```

```
Out[6]: range(0, 100, 10)
```

```
In [7]: list(range(0,100,10))
```

```
Out[7]: [0, 10, 20, 30, 40, 50, 60, 70, 80, 90]
```

```
In [8]: range(0,100,20)
```

```
Out[8]: range(0, 100, 20)
```

```
In [9]: list(range(0,100,20)) # it expected only take 3 arguments but we canot pass 4 argum
```

```
Out[9]: [0, 20, 40, 60, 80]
```

```
In [10]: range(0,100,20,5)
```

```
-----  
TypeError                                 Traceback (most recent call last)  
Cell In[10], line 1  
----> 1 range(0,100,20,5)
```

```
TypeError: range expected at most 3 arguments, got 4
```