

```
#Data Structure ----- *List *Tuple *Set *Dict *Range
```

#1.List

```
In [2]: l = []      #list is define in '[]'  
l
```

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

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

```
Out[3]: list
```

```
In [4]: len(l)
```

```
Out[4]: 0
```

```
In [5]: list = [2,8,4,6]  
list
```

```
Out[5]: [2, 8, 4, 6]
```

*l.append() -----add the element at the end

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

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

```
In [7]: l.append(10)  
l
```

```
Out[7]: [10]
```

```
In [8]: l.append(11)  
l.append(12)  
l.append(13)  
l.append(14)  
l
```

```
Out[8]: [10, 11, 12, 13, 14]
```

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

```
Out[9]: 5
```

```
In [10]: l.append(15)  
l
```

```
Out[10]: [10, 11, 12, 13, 14, 15]
```

```
In [87]: l.append([1,2,3])  
l
```

```
Out[87]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb', 'Mar', 20, 3, [1, 2, 3]]
```

```
In [11]: l1 = [12,3.5,'one',False,2+4j]      # contain more than one datatype
l1
```

```
Out[11]: [12, 3.5, 'one', False, (2+4j)]
```

```
In [12]: print(l)
print(l1)
```

```
[10, 11, 12, 13, 14, 15]
[12, 3.5, 'one', False, (2+4j)]
```

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

```
6
5
```

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

```
2019386528768
2019386508480
```

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

```
Out[17]: [12, 3.5, 'one', False, (2+4j)]
```

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

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

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

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

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

```
[10, 11, 12, 13, 14, 15]
[12, 3.5, 'one', False, (2+4j)]
[12, 3.5, 'one', False, (2+4j)]
```

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

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

*indexing

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

```
Out[22]: [10, 11, 12, 13, 14, 15]
```

```
In [23]: l[0]
```

```
Out[23]: 10
```

```
In [24]: l[-1]
```

Out[24]: 15

In [26]:
print(l[4])
print(l[2])
print(l[5])14
12
15In [27]:
print(l[-6])
print(l[-4])
print(l[-3])10
12
13

In [28]: 1

Out[28]: [10, 11, 12, 13, 14, 15]

In [29]: l[0] = 100
l

Out[29]: [100, 11, 12, 13, 14, 15]

In [30]: l[-1] = 'Feb'
l

Out[30]: [100, 11, 12, 13, 14, 'Feb']

In [31]: l.append('mar')
l

Out[31]: [100, 11, 12, 13, 14, 'Feb', 'mar']

In [32]: l.append(20)
l

Out[32]: [100, 11, 12, 13, 14, 'Feb', 'mar', 20]

In [33]: 12

Out[33]: [12, 3.5, 'one', False, (2+4j)]

In [34]: len(12)

Out[34]: 5

*l.clear() ----- clear the value from the list but empty list

In [35]: 12.clear()
12

Out[35]: []

In [36]: len(12)

```
Out[36]: 0
```

```
In [37]: 1
```

```
Out[37]: [100, 11, 12, 13, 14, 'Feb', 'mar', 20]
```

*l.count() -----count the frequency of element

```
In [38]: l.count(100)
```

```
Out[38]: 1
```

```
In [39]: l[2] = 20  
l
```

```
Out[39]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [40]: l.count(20)
```

```
Out[40]: 2
```

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

```
8
```

```
0
```

*list membership

```
In [42]: 1
```

```
Out[42]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [43]: 11 in l
```

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

```
In [44]: 200 in l
```

```
Out[44]: False
```

```
In [45]: print(l)  
print(l1)  
print(l2)
```

```
[100, 11, 20, 13, 14, 'Feb', 'mar', 20]  
[12, 3.5, 'one', False, (2+4j)]  
[]
```

*l.extend() -----merge list values to another list

```
In [46]: l2.extend(l)  
l2
```

```
Out[46]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [47]: l2.extend(l1)  
l2
```

```
Out[47]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [48]: l1.extend(l2)
l1
```

```
Out[48]: [12,
 3.5,
 'one',
 False,
 (2+4j),
 100,
 11,
 20,
 13,
 14,
 'Feb',
 'mar',
 20,
 12,
 3.5,
 'one',
 False,
 (2+4j)]
```

```
In [49]: len(l1)
```

```
Out[49]: 18
```

*l.index() -----it gives index value of the element

```
In [50]: l
```

```
Out[50]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [52]: l.index('Feb')
```

```
Out[52]: 5
```

```
In [53]: l.index(20)
```

```
Out[53]: 2
```

*Forward slicing

```
In [54]: l
```

```
Out[54]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [55]: l[:]
```

```
Out[55]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [56]: l2
```

```
Out[56]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [57]: l2[:]
```

```
Out[57]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [58]: l2[0:8]
```

```
Out[58]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [59]: 12[3:10]
```

```
Out[59]: [13, 14, 'Feb', 'mar', 20, 12, 3.5]
```

```
In [60]: 12
```

```
Out[60]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [61]: 12[::-2]
```

```
Out[61]: [100, 20, 14, 'mar', 12, 'one', (2+4j)]
```

```
In [62]: 12[::-5]
```

```
Out[62]: [100, 'Feb', 'one']
```

```
In [63]: 12[2:10:3]
```

```
Out[63]: [20, 'Feb', 12]
```

```
In [64]: 12[0:11:4]
```

```
Out[64]: [100, 14, 12]
```

*Backward slicing

```
In [65]: 12
```

```
Out[65]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [66]: 12[-1]
```

```
Out[66]: (2+4j)
```

```
In [67]: 12[-3]
```

```
Out[67]: 'one'
```

```
In [68]: 12[:-9]
```

```
Out[68]: [100, 11, 20, 13]
```

```
In [69]: 12[:-3]
```

```
Out[69]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5]
```

```
In [71]: 12[-3:]
```

```
Out[71]: ['one', False, (2+4j)]
```

```
In [72]: 12[3:-3]
```

```
Out[72]: [13, 14, 'Feb', 'mar', 20, 12, 3.5]
```

```
In [73]: 12[::-1]
```

```
Out[73]: [(2+4j), False, 'one', 3.5, 12, 20, 'mar', 'Feb', 14, 13, 20, 11, 100]
```

```
In [74]: 12[:-7]
```

```
Out[74]: [100, 11, 20, 13, 14, 'Feb']
```

```
In [75]: 12[::-3]
```

```
Out[75]: [(2+4j), 3.5, 'mar', 13, 100]
```

*l.remove() ----- remove element 1st occurrence

```
In [76]: print(l)
print(l1)
print(l2)
```

```
[100, 11, 20, 13, 14, 'Feb', 'mar', 20]
[12, 3.5, 'one', False, (2+4j), 100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5,
'one', False, (2+4j)]
[100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [77]: l1.remove(2+4j)
```

```
In [78]: print(l1)
```

```
[12, 3.5, 'one', False, 100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', Fa
lse, (2+4j)]
```

```
In [80]: 12.remove('Feb')
12
```

```
Out[80]: [100, 11, 20, 13, 14, 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

*l.insert() -----

```
In [81]: print(l)
print(l1)
print(l2)
```

```
[100, 11, 20, 13, 14, 'Feb', 'mar', 20]
[12, 3.5, 'one', False, 100, 11, 20, 13, 14, 'Feb', 'mar', 20, 12, 3.5, 'one', Fa
lse, (2+4j)]
[100, 11, 20, 13, 14, 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [82]: 1
```

```
Out[82]: [100, 11, 20, 13, 14, 'Feb', 'mar', 20]
```

```
In [84]: l.insert(3,38) # insert 38 in 3rd index
1
```

```
Out[84]: [100, 11, 20, 38, 13, 14, 'Feb', 'mar', 20, 3]
```

```
In [85]: l.insert(6,'Jan') # insert 38 in 3rd index
1
```

```
Out[85]: [100, 11, 20, 38, 13, 14, 'Jan', 'Feb', 'mar', 20, 3]
```

```
In [86]: l.insert(4, 35)  
l
```

```
Out[86]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb', 'mar', 20, 3]
```

```
*l.pop() -----
```

```
In [88]: l
```

```
Out[88]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb', 'mar', 20, 3, [1, 2, 3]]
```

```
In [89]: l.pop()
```

```
Out[89]: [1, 2, 3]
```

```
In [90]: l.pop()
```

```
Out[90]: 3
```

```
In [91]: l
```

```
Out[91]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb', 'mar', 20]
```

```
In [92]: l.pop()
```

```
Out[92]: 20
```

```
In [93]: l
```

```
Out[93]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb', 'mar']
```

```
In [94]: l.pop()
```

```
Out[94]: 'mar'
```

```
In [95]: l
```

```
Out[95]: [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb']
```

```
In [97]: 12
```

```
Out[97]: [100, 11, 20, 13, 14, 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [100...]: 12.pop(False)
```

```
Out[100...]: 100
```

```
In [101...]: 12
```

```
Out[101...]: [11, 20, 13, 14, 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [102...]: 12.pop(True)
```

```
Out[102...]: 20
```

```
In [103...]: 13 = [100, 2, 39, 34, 20, 1000]
```

13

Out[103... [100, 2, 39, 34, 20, 1000]

In [104... 13.sort(reverse=True) # hyperparameter
13

Out[104... [1000, 100, 39, 34, 20, 2]

In [105... 13.sort(reverse=False)
13

Out[105... [2, 20, 34, 39, 100, 1000]

In [107... 14 = [10, 2, 80, 34, 20, 300]
14

Out[107... [10, 2, 80, 34, 20, 300]

In [108... 14.sort() #parameter
14

Out[108... [2, 10, 20, 34, 80, 300]

In [109... 15 = ['m', 'b', 'z', 'a']
15

Out[109... ['m', 'b', 'z', 'a']

In [111... 15.sort()
15

Out[111... ['a', 'b', 'm', 'z']

*reverse() -----

In [114... 11 = [100, 2, 39, 34, 20, 1000]
11

Out[114... [100, 2, 39, 34, 20, 1000]

In [115... 11.sort()
11

Out[115... [2, 20, 34, 39, 100, 1000]

In [116... 11.sort(reverse=True)
11

Out[116... [1000, 100, 39, 34, 20, 2]

In [117... 15 = ['m', 'b', 'z', 'a']
15

Out[117... ['m', 'b', 'z', 'a']

In [118... 15.reverse()

```
In [119... 15
```

```
Out[119... ['a', 'z', 'b', 'm']
```

```
In [120... 1
```

```
Out[120... [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb']
```

```
In [121... 1[7]
```

```
Out[121... 'Jan'
```

```
In [122... print(l[7][0])
         print(l[7][1])
         print(l[7][2])
```

```
J
a
n
```

```
In [124... 1
```

```
Out[124... [100, 11, 20, 38, 35, 13, 14, 'Jan', 'Feb']
```

```
In [125... 12
```

```
Out[125... [11, 13, 14, 'mar', 20, 12, 3.5, 'one', False, (2+4j)]
```

```
In [127... for i in l:
           print(i)
```

```
100
11
20
38
35
13
14
Jan
Feb
```

```
In [128... 13
```

```
Out[128... [2, 20, 34, 39, 100, 1000]
```

```
In [129... for i in l3:
           print(i)
```

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

```
In [130... for i in enumerate(l3):
           print(i)
```

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

In [131... `L1 = [1,2,3,4,0]`

In [132... `all(L1)`

Out[132... `False`

In [133... `any(L1)`

Out[133... `True`

In [134... `L2 = [1,2,3,4]`

In [135... `all(L2)`

Out[135... `True`

In [136... `any(L2)`

Out[136... `True`

#2.Tuple

In [157... *#once we enter a value in tuple no one can change that
#only 2 function work --.index(),.count(),,remove is not allowed*

In [138... `t = ()`
`t`

Out[138... `()`

In [139... `type(t)`

Out[139... `tuple`

In [140... `t = (15,8,65)`
`t`

Out[140... `(15, 8, 65)`

In [141... `t1 = (10,20,2.2,'ten', True, 1+2j,20)`
`t1`

Out[141... `(10, 20, 2.2, 'ten', True, (1+2j), 20)`

In [142... `t1.count(20)`

Out[142... `2`

```
In [143...]: t1.index(20)
```

```
Out[143...]: 1
```

```
In [144...]: print(t)
print(t1)
```

```
(15, 8, 65)
(10, 20, 2.2, 'ten', True, (1+2j), 20)
```

```
In [145...]: print(len(t))
print(len(t1))
```

```
3
```

```
7
```

```
In [146...]: t
```

```
Out[146...]: (15, 8, 65)
```

```
In [147...]: t[0]
```

```
Out[147...]: 15
```

```
In [149...]: bank_account = (1234, 'cizp45yi', 10000)
bank_account
```

```
Out[149...]: (1234, 'cizp45yi', 10000)
```

```
In [150...]: t
```

```
Out[150...]: (15, 8, 65)
```

```
In [154...]: t2 = t * 3
t2
```

```
Out[154...]: (15, 8, 65, 15, 8, 65, 15, 8, 65)
```

```
In [155...]: for i in t:
    print(i)
```

```
15
```

```
8
```

```
65
```

```
In [156...]: for i in enumerate(t):
    print(i)
```

```
(0, 15)
```

```
(1, 8)
```

```
(2, 65)
```

#3.Set

```
In [9]: s = {}
```

```
Out[9]: {}
```

```
In [159]: type(s)
```

```
Out[159]: dict
```

```
In [11]: s1 = set()  
s1
```

```
Out[11]: set()
```

```
In [2]: s2 = {32, 48, 54, 38, 22, 45, 11}  
s2
```

```
Out[2]: {11, 22, 32, 38, 45, 48, 54}
```

```
In [3]: type(s2)
```

```
Out[3]: set
```

```
In [4]: s3 = s2.copy()  
s3
```

```
Out[4]: {11, 22, 32, 38, 45, 48, 54}
```

```
In [5]: s3.add(4.8)      #add()  
s3
```

```
Out[5]: {4.8, 11, 22, 32, 38, 45, 48, 54}
```

```
In [6]: s3.add('mit')  
s3
```

```
Out[6]: {11, 22, 32, 38, 4.8, 45, 48, 54, 'mit'}
```

```
In [7]: s3.add(2+3j)  
s3.add(False)  
s3
```

```
Out[7]: {(2+3j), 11, 22, 32, 38, 4.8, 45, 48, 54, False, 'mit'}
```

```
In [12]: print(s)  
print(s1)  
print(s2)  
print(s3)
```

```
{}  
set()  
{32, 48, 54, 38, 22, 11, 45}  
{False, 4.8, 11, (2+3j), 22, 32, 38, 45, 'mit', 48, 54}
```

```
In [13]: type(s)
```

```
Out[13]: dict
```

```
In [14]: s3
```

```
Out[14]: {(2+3j), 11, 22, 32, 38, 4.8, 45, 48, 54, False, 'mit'}
```

```
In [15]: s3.remove(2+3j)      #remove()
```

```
In [16]: s3
```

```
Out[16]: {11, 22, 32, 38, 4.8, 45, 48, 54, False, 'mit'}
```

```
In [17]: s3.discard(2000)      #discard()
s3
```

```
Out[17]: {11, 22, 32, 38, 4.8, 45, 48, 54, False, 'mit'}
```

```
In [18]: s3.discard(38)
s3
```

```
Out[18]: {11, 22, 32, 4.8, 45, 48, 54, False, 'mit'}
```

```
In [20]: s3
```

```
Out[20]: {11, 22, 32, 4.8, 45, 48, 54, 'mit'}
```

```
In [23]: s3.pop()
```

```
Out[23]: 22
```

```
In [24]: s3
```

```
Out[24]: {32, 45, 48, 54, 'mit'}
```

```
In [27]: s3.pop()      # pop() takes no argument
```

```
Out[27]: 45
```

```
In [28]: s3
```

```
Out[28]: {48, 54, 'mit'}
```

```
In [29]: 54 in s3      #membership
```

```
Out[29]: True
```

```
In [30]: 33 in s3
```

```
Out[30]: False
```

set operation

```
In [31]: a = {1,2,3,4,5}
b = {4,5,6,7,8}
c = {8,9,10}
```

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

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

```
In [35]: a.union(b) # union() operation
```

```
Out[35]: {1, 2, 3, 4, 5, 6, 7, 8}
```

```
In [36]: a.union(b,c)
```

```
Out[36]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
In [37]: a = {1,2,3,5,6}  
b = {6,7,8,9,10}  
c = {11,12,10}
```

```
In [38]: a | b # | means union
```

```
Out[38]: {1, 2, 3, 5, 6, 7, 8, 9, 10}
```

```
In [39]: b | c
```

```
Out[39]: {6, 7, 8, 9, 10, 11, 12}
```

```
In [40]: a | b | c
```

```
Out[40]: {1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12}
```

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

```
Out[41]: {1, 2, 3, 5, 6, 10, 11, 12}
```

```
# intersection()
```

```
In [43]: a = {1,2,3,5,6}  
b = {6,7,8,9,10}  
c = {11,12,10}
```

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

```
Out[44]: {6}
```

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

```
Out[45]: {10}
```

```
In [46]: a & b # & means intersection
```

```
Out[46]: {6}
```

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

```
Out[47]: {10}
```

```
In [48]: a & c
```

```
Out[48]: set()
```

```
# difference()
```

```
In [49]: a = {1,2,3,4,5}  
b = {4,5,6,7,8}  
c = {8,9,10}
```

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

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

```
In [51]: b.difference(a)
```

```
Out[51]: {6, 7, 8}
```

```
In [52]: a - b          # - means difference
```

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

```
In [53]: a - c
```

```
Out[53]: {1, 2, 3, 4, 5}
```

```
In [54]: b - c
```

```
Out[54]: {4, 5, 6, 7}
```

```
In [55]: a - b - c
```

```
Out[55]: {1, 2, 3}
```

```
# symmetric_difference()
```

```
In [56]: print(a)  
print(b)  
print(c)
```

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

```
In [57]: a.symmetric_difference(b)
```

```
Out[57]: {1, 2, 3, 6, 7, 8}
```

```
In [58]: a.symmetric_difference(c)
```

```
Out[58]: {1, 2, 3, 4, 5, 8, 9, 10}
```

```
In [59]: a ^ b      # ^ means symmetric_difference
```

```
Out[59]: {1, 2, 3, 6, 7, 8}
```

```
In [60]: b ^ c
```

```
Out[60]: {4, 5, 6, 7, 9, 10}
```

```
In [61]: a.symmetric_difference_update(b)
```

```
In [62]: a
```

```
Out[62]: {1, 2, 3, 6, 7, 8}
```

```
In [63]: b.symmetric_difference_update(c)
```

```
In [64]: b
```

```
Out[64]: {4, 5, 6, 7, 9, 10}
```

superset,subset,disjoint

```
In [65]: s4 = {1,2,3,4,5,6,7,8,9}  
s5 = {3,4,5,6,7,8}  
s6 = {10,20,30,40}
```

```
In [66]: s4.issuperset(s5)
```

```
Out[66]: True
```

```
In [67]: s4.issuperset(s6)
```

```
Out[67]: False
```

```
In [68]: s5.issuperset(s4)
```

```
Out[68]: False
```

```
In [69]: s4 = {1,2,3,4,5,6,7,8,9}  
s5 = {3,4,5,6,7,8}  
s6 = {10,20,30,40}
```

```
In [70]: s6.isdisjoint(s4)
```

```
Out[70]: True
```

```
In [71]: s6.isdisjoint(s5)
```

```
Out[71]: True
```

```
In [72]: s5.isdisjoint(s6)
```

```
Out[72]: True
```

```
In [73]: s4.isdisjoint(s5)
```

```
Out[73]: False
```

```
In [74]: s4 = {1,2,3,4,5,6,7,8,9}  
s5 = {3,4,5,6,7,8}  
s6 = {10,20,30,40}
```

```
In [75]: s6.issubset(s5)
```

```
Out[75]: False
```

```
In [76]: s6.issubset(s4)
```

```
Out[76]: False
```

```
In [77]: s4.issubset(s5)
```

```
Out[77]: False
```

```
In [78]: s5.issubset(s4)
```

```
Out[78]: True
```

#4.Dictionary

```
In [79]: d = {}
```

```
d
```

```
Out[79]: {}
```

```
In [80]: type(d)
```

```
Out[80]: dict
```

```
In [82]: d1 = {1 : 'one', 2: 'two', 3: 'three', 'four': 4, }  
d1
```

```
Out[82]: {1: 'one', 2: 'two', 3: 'three', 'four': 4}
```

```
In [84]: d2 = d1.copy()      #copy()  
d2
```

```
Out[84]: {1: 'one', 2: 'two', 3: 'three', 'four': 4}
```

```
In [85]: d1.items()      #items()
```

```
Out[85]: dict_items([(1, 'one'), (2, 'two'), (3, 'three'), ('four', 4)])
```

```
In [86]: len(d1)
```

```
Out[86]: 4
```

```
In [87]: d1.keys()
```

```
Out[87]: dict_keys([1, 2, 3, 'four'])
```

```
In [88]: d1.values()
```

```
Out[88]: dict_values(['one', 'two', 'three', 4])
```

```
In [89]: d1
```

```
Out[89]: {1: 'one', 2: 'two', 3: 'three', 'four': 4}
```

```
In [94]: d1[2]      #indexing with keys
```

```
Out[94]: 'two'
```

```
In [95]: d1[1]
```

```
Out[95]: 'one'
```

```
In [96]: d1
```

```
Out[96]: {1: 'one', 2: 'two', 3: 'three', 'four': 4}
```

```
In [97]: 100 in d1
```

```
Out[97]: False
```

```
In [98]: 2 in d1
```

```
Out[98]: True
```

```
In [99]: 'four' in d1
```

```
Out[99]: True
```

```
In [100...]: 'ten' in d1
```

```
Out[100...]: False
```

```
In [102...]: d1.pop(3) # at Least one argument
```

```
Out[102...]: 'three'
```

```
In [103...]: d1.pop(1)
```

```
Out[103...]: 'one'
```

#5.Range

```
In [104...]: range(10)
```

```
Out[104...]: range(0, 10)
```

```
In [105...]: range(100)
```

```
Out[105...]: range(0, 100)
```

```
In [106...]: range(20,40)
```

```
Out[106...]: range(20, 40)
```

```
In [107...]: range(10,30,5)
```

```
Out[107...]: range(10, 30, 5)
```

```
In [115...]: a1 = list(range(10))
```

```
a1
```

```
Out[115... [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [116... list(range(20,40))
```

```
Out[116... [20,  
           21,  
           22,  
           23,  
           24,  
           25,  
           26,  
           27,  
           28,  
           29,  
           30,  
           31,  
           32,  
           33,  
           34,  
           35,  
           36,  
           37,  
           38,  
           39]
```

```
In [117... list(range(10,30,5))
```

```
Out[117... [10, 15, 20, 25]
```

```
In [118... r = range(10, 20, 5)  
r
```

```
Out[118... range(10, 20, 5)
```

```
In [119... for i in r:  
        print(i)
```

```
10  
15
```

```
In [120... for i in a1:  
        print(i)
```

```
0  
1  
2  
3  
4  
5  
6  
7  
8  
9
```

```
In [ ]:
```