In [4]:
```
set1 = {1,2,3,4,5,6,7}
set1
```

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

In [5]:
```
type(set1)
```

Out[5]:
```
set
```

In [6]:
```
len(set1)
```

Out[6]:
```
7
```

In [7]:
```
set1 ={1,232,4323,4532,4532,23333}
```

In [9]:
```
set2={'Rohit','Mohit','John'}
set2
```

Out[9]:
```
{'John', 'Mohit', 'Rohit'}
```

In [10]:
```
set3={10,20,"Rohit",(11,22,33)}
set3
```

Out[10]:
```
{(11, 22, 33), 10, 20, 'Rohit'}
```

In [13]:
```
set4 = {10,23,34,'Rohit',[12,34,56]}  # set doesnot allow utabele items
set4
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[13], line 1
----> 1 set4 = {10,23,34,'Rohit',[12,34,56]}  # set doesnot allow utabele items
      2 set4

TypeError: unhashable type: 'list'
```

Loop through a set

In [14]:
```
myset = {'one','two','three','four','five','six','seven','eight'}
for i in myset:
    print(i)
```

```
eight
two
five
four
one
three
seven
six
```

In [16]:
```
for i in enumerate(myset):
    print(i)
```

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

set Membership

In [17]:
```
myset
```

Out[17]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [18]:
```
'one' in myset # check one is present or not
```

Out[18]: True

In [24]:
```
'ten' in myset
```

Out[24]: False

In [25]:
```
if 'three' in myset: # Check if 'three' exist in the set
   print('Three is present in the set')
else:
  print('Three is not present in the set')
```

```
Three is present in the set
```

In [26]:
```
if 'eleven' in myset: # Check if 'eleven' exist in the list
  print('eleven is present in the set')
else:
  print('eleven is not present in the set')
```

```
eleven is not present in the set
```

Add & Remove Items

In [27]:
```
myset
```

Out[27]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

In [28]:
```
myset.add("nine")
```

In [29]:
```
myset
```

Out[29]: {'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}

In [30]:
```
myset.update(['ten','eleven','twelven'])
myset
```

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

In [31]:
```
myset.remove('nine') # remove item in a set using remove() method
myset
```

Out[31]:
```
{'eight',
 'eleven',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'ten',
 'three',
 'twelven',
 'two'}
```

In [32]:
```
myset.discard('TEN') # remove item from a set using discard() method
myset
```

Out[32]:
```
{'eight',
 'eleven',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'ten',
 'three',
 'twelven',
 'two'}
```

In [33]:
```
myset.clear() # Delete all items in a set
myset
```

Out[33]:
```
set()
```

In [34]:
```
del myset # Delete the set object
myset
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[34], line 2
      1 del myset # Delete the set object
----> 2 myset

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

In [35]:
```
myset = {'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight'}
myset
```

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

In [36]:
```
myset1 = myset
myset1
```

Out[36]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [38]:
```
id(myset) , id(myset1) # the addresh are same
```

Out[38]:
```
(2690392715360, 2690392715360)
```

Set Operation

In [39]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
C = {8,9,10}
```

In [40]:
```
A.union(B)
```

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

In [41]:
```
A|B
```

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

In [42]:
```
A.union(B, C)
```

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

In [44]:
```
A|B|C
```

Out[44]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

Intersection

In [46]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [47]:
```
A.intersection(B)
```

Out[47]:
```
{4, 5}
```

In [48]:
```
A&B
```

Out[48]: {4, 5}

Difference

In [49]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [50]:
```
A.difference(B)
```

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

In [51]:
```
A - B
```

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

In [52]:
```
B.difference(A)
```

Out[52]: {6, 7, 8}

In [53]:
```
B- A
```

Out[53]: {6, 7, 8}

Symmetric Difference

In [55]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

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

Out[56]: {1, 2, 3, 6, 7, 8}

In [57]:
```
A ^ B
```

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

Subset Superset & Disjoin

In [58]:
```
A = {1,2,3,4,5,6,7,8,9}
B = {3,4,5,6,7,8}
C = {10,20,30,40}
```

In [59]:
```
B.issubset(A)
```

Out[59]: True

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

Out[60]: True

In [61]:
```
C.isdisjoint(A)
```

Out[61]: True

In [62]:
```
B.isdisjoint(A)
```

Out[62]: False

Other Builtin functions

In [63]:
```
A
```

Out[63]: {1, 2, 3, 4, 5, 6, 7, 8, 9}

In [64]:
```
sum(A)
```

Out[64]: 45

In [65]:
```
max(A)
```

Out[65]: 9

In [66]:
```
min(A)
```

Out[66]: 1

In [67]:
```
len(A)
```

Out[67]: 9

In [68]:
```
list(enumerate(A))
```

Out[68]: [(0, 1), (1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8), (8, 9)]

In [69]:
```
D= sorted(A,reverse=True)
D
```

Out[69]: [9, 8, 7, 6, 5, 4, 3, 2, 1]

Dictionary

In [70]:
```
mydict = dict() # empty dictionary
mydict
```

Out[70]: {}

In [71]:
```
mydict = {} # empty dictionary
mydict
```

Out[71]: {}

In [72]:
```
mydict = {1:'one' , 2:'two' , 3:'three'}
mydict
```

Out[72]: {1: 'one', 2: 'two', 3: 'three'}

In [73]:
```
mydict = dict({1:'one' , 2:'two' , 3:'three'})
mydict
```

Out[73]:
```
{1: 'one', 2: 'two', 3: 'three'}
```

In [74]:
```
mydict.keys()
```

Out[74]:
```
dict_keys([1, 2, 3])
```

In [75]:
```
mydict.values()
```

Out[75]:
```
dict_values(['one', 'two', 'three'])
```

In [76]:
```
mydict.items()
```

Out[76]:
```
dict_items([(1, 'one'), (2, 'two'), (3, 'three')])
```

In [77]:
```
mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria']}
mydict
```

Out[77]:
```
{1: 'one', 2: 'two', 'A': ['asif', 'john', 'Maria']}
```

In [78]:
```
keys = {'a' , 'b' , 'c' , 'd'}
mydict3 = dict.fromkeys(keys) # Create a dictionary from a sequence of keys
mydict3
```

Out[78]:
```
{'b': None, 'd': None, 'c': None, 'a': None}
```

In [79]:
```
keys = {'a' , 'b' , 'c' , 'd'}
value = 10
mydict3 = dict.fromkeys(keys , value) # Create a dictionary from a sequence of
mydict3
```

Out[79]:
```
{'b': 10, 'd': 10, 'c': 10, 'a': 10}
```

In [80]:
```
keys = {'a' , 'b' , 'c' , 'd'}
value = [10,20,30]
mydict3 = dict.fromkeys(keys , value) # Create a dictionary from a sequence of
mydict3
```

Out[80]:
```
{'b': [10, 20, 30], 'd': [10, 20, 30], 'c': [10, 20, 30], 'a': [10, 20, 30]}
```

In [81]:
```
value.append(40)
mydict3
```

Out[81]:
```
{'b': [10, 20, 30, 40],
 'd': [10, 20, 30, 40],
 'c': [10, 20, 30, 40],
 'a': [10, 20, 30, 40]}
```

Accessing Items

In [82]:
```
dict1 ={1:'one' , 2:'two' , 3:'three' , 4:'four'}
dict1
```

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

In [83]:
```
dict1[1]
```

Out[83]: 'one'

In [84]:
```
dict1.get(1)
```

Out[84]: 'one'

In [85]:
```
mydict = {'Name':'Rohit' , 'ID': 205 , 'DOB': 2003 , 'job' :'Analyst'}
mydict
```

Out[85]: {'Name': 'Rohit', 'ID': 205, 'DOB': 2003, 'job': 'Analyst'}

In [86]:
```
mydict['Name']
```

Out[86]: 'Rohit'

In [88]:
```
mydict.get('job')
```

Out[88]: 'Analyst'

Add, Remove & Change Items

In [89]:
```
mydict
```

Out[89]: {'Name': 'Rohit', 'ID': 205, 'DOB': 2003, 'job': 'Analyst'}

In [90]:
```
mydict['DOB'] = 2004 # Changing Dictionary Items
mydict['Address'] = 'Delhi'
mydict
```

Out[90]: {'Name': 'Rohit', 'ID': 205, 'DOB': 2004, 'job': 'Analyst', 'Address': 'Delhi'}

In [92]:
```
dict1 = {'DOB':2004}
mydict.update(dict1)
mydict
```

Out[92]: {'Name': 'Rohit', 'ID': 205, 'DOB': 2004, 'job': 'Analyst', 'Address': 'Delhi'}

In [95]:
```
mydict['Job'] = 'Data Scientist' # Adding items in the dictionary
mydict
```

Out[95]:
```
{'Name': 'Rohit',
 'ID': 205,
 'DOB': 2004,
 'job': 'Analyst',
 'Address': 'Delhi',
 'Job': 'Data Scientist'}
```

In [96]:
```
mydict.pop('Job') # Removing items in the dictionary using Pop method
mydict
```

Out[96]: {'Name': 'Rohit', 'ID': 205, 'DOB': 2004, 'job': 'Analyst', 'Address': 'Delhi'}

In [97]:
```
mydict.popitem()
```

Out[97]:
```
('Address', 'Delhi')
```

In [98]:
```
mydict
```

Out[98]:
```
{'Name': 'Rohit', 'ID': 205, 'DOB': 2004, 'job': 'Analyst'}
```

In [99]:
```
del[mydict['ID']] # Removing item using del method
mydict
```

Out[99]:
```
{'Name': 'Rohit', 'DOB': 2004, 'job': 'Analyst'}
```

In [100…
```
mydict.clear() # Delete all items of the dictionary using clear method
mydict
```

Out[100…
```
{}
```

Copy Dictionary

In [102…
```
mydict = {'Name':'Rohit' , 'ID': 12345 , 'DOB': 2003 , 'Address' : 'Odisha'}
mydict
```

Out[102…
```
{'Name': 'Rohit', 'ID': 12345, 'DOB': 2003, 'Address': 'Odisha'}
```

In [103…
```
mydict1 = mydict
```

In [104…
```
id(mydict) , id(mydict1)
```

Out[104…
```
(2690391365632, 2690391365632)
```

In [105…
```
mydict2 = mydict.copy()
```

In [106…
```
id(mydict2)
```

Out[106…
```
2690404144384
```

In [107…
```
mydict['Address'] = 'Mumbai'
```

In [108…
```
mydict
```

Out[108…
```
{'Name': 'Rohit', 'ID': 12345, 'DOB': 2003, 'Address': 'Mumbai'}
```

In [109…
```
mydict1
```

Out[109…
```
{'Name': 'Rohit', 'ID': 12345, 'DOB': 2003, 'Address': 'Mumbai'}
```

In [110…
```
mydict2
```

Out[110…
```
{'Name': 'Rohit', 'ID': 12345, 'DOB': 2003, 'Address': 'Odisha'}
```

Loop through a Dictionary

In [112…
```
mydict1 = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Odisha' }
mydict1
```

Out[112…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Odisha'}
```

In [114…
```
for i in mydict1:
 print(i , ':' , mydict1[i])
```

```
Name : Asif
ID : 12345
DOB : 1991
Address : Odisha
```

Dictionary Membership

In [115…
```
mydict1 = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Odisha' }
mydict1
```

Out[115…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Odisha'}
```

In [116…
```
'Name' in mydict1
```

Out[116…
```
True
```

In [117…
```
'Asif' in mydict1
```

Out[117…
```
False
```

In [ ]: