In [25]:
```
tup1=()
```

In [26]:
```
tup2=(10,20,30)
```

In [27]:
```
tup3=(10.77,30.60,60.89)
```

In [28]:
```
tup4=('one','two','three')
```

In [29]:
```
tup5=('asif',25,(50,100),(150,90))
```

In [30]:
```
tup6=(100,'asif',17.765)
```

In [31]:
```
tup7=('asif',25,[50,100],[150,90],{'david','kamal'},(99,22,33))
```

In [32]:
```
len(tup7)
```

Out[32]: 6

In [33]:
```
tup2[0]
```

Out[33]: 10

In [34]:
```
tup4[0]
```

Out[34]: 'one'

In [35]:
```
tup4[0][0]
```

Out[35]: 'o'

In [36]:
```
tup4[-1]
```

Out[36]: 'three'

In [37]:
```
tup5[-1]
```

Out[37]: (150, 90)

In [38]:
```
mytup=('nine','eight','seven','six','five','three','four','two','one')
```

In [39]:
```
mytup[0:3]
```

Out[39]: ('nine', 'eight', 'seven')

In [40]:
```
mytup[2:5]
```

Out[40]: ('seven', 'six', 'five')

In [41]:
```
mytup[:-3]
```

Out[41]: ('nine', 'eight', 'seven', 'six', 'five', 'three')

In [42]:
```
mytup[:2]
```

Out[42]: ('nine', 'eight')

In [43]:
```
mytup[:3]
```

Out[43]: ('nine', 'eight', 'seven')

In [44]:
```
mytup[-3:]
```

Out[44]: ('four', 'two', 'one')

In [45]:
```
mytup[-2:]
```

Out[45]: ('two', 'one')

In [46]:
```
mytup[:]
```

Out[46]: ('nine', 'eight', 'seven', 'six', 'five', 'three', 'four', 'two', 'one')

In [47]:
```
mytup
```

Out[47]: ('nine', 'eight', 'seven', 'six', 'five', 'three', 'four', 'two', 'one')

In [24]:
```
del mytup[0]
#tuples are immutables
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[24], line 1
----> 1 del mytup[0]

TypeError: 'tuple' object doesn't support item deletion
```

In [48]:
```
mytup[0]=1
#tuples are immutables
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[48], line 1
----> 1 mytup[0]=1

TypeError: 'tuple' object does not support item assignment
```

In [49]:
```
del mytup #deleting entire tuple object is possible
```

In [50]:
```
mytup=('nine','eight','seven','six','five','three','four','two','one')
mytup
```

Out[50]:
```
('nine', 'eight', 'seven', 'six', 'five', 'three', 'four', 'two', 'one')
```

In [51]:
```
for i in mytup:
    print(i)
```

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

In [52]:
```
for i in enumerate(mytup):
    print(i)
```

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

In [53]:
```
mytup23=('even','odd','win','fer','yash','my','true','false')
```

In [54]:
```
mytup23.count('even')
```

Out[54]:
```
1
```

In [55]:
```
mytup
```

Out[55]:
```
('nine', 'eight', 'seven', 'six', 'five', 'three', 'four', 'two', 'one')
```

In [56]:
```
'nine'in mytup
```

Out[56]:
```
True
```

In [57]:
```
'even'in mytup
```

Out[57]:
```
False
```

In [58]:
```
if 'nine' in  mytup:
   print('nine is present in the tuples')
else:
   print('nine is not present in the tuple')
```

```
nine is present in the tuples
```

In [59]:
```
mytup23
```

Out[59]:
```
('even', 'odd', 'win', 'fer', 'yash', 'my', 'true', 'false')
```

In [60]:
```
mytup23.index('odd')
```

Out[60]: 1

In [61]:
```
mytup23.index('yash')
```

Out[61]: 4

In [62]:
```
mytup24=(43,67,99,12,6,90,67)
```

In [63]:
```
sorted(mytup24)
```

Out[63]: [6, 12, 43, 67, 67, 90, 99]

In [64]:
```
sorted(mytup24,reverse=True)
```

Out[64]: [99, 90, 67, 67, 43, 12, 6]

*******SEt**********

In [65]:
```
myset={1,2,3,4,5}
myset
```

Out[65]: {1, 2, 3, 4, 5}

In [66]:
```
len(myset)
```

Out[66]: 5

In [67]:
```
my_set={1,1,2,2,3,4,5,5}
my_set
```

Out[67]: {1, 2, 3, 4, 5}

In [68]:
```
myset1={1.79,2.08,3.99,4.56,5.45,7.89}
myset1
```

Out[68]: {1.79, 2.08, 3.99, 4.56, 5.45, 7.89}

In [69]:
```
myset2={'asif','yash','umesh'}
myset2
```

Out[69]:
```
{'asif', 'umesh', 'yash'}
```

In [70]:
```
myset3={10,20,'harsh',(11,22,33)}
myset3
```

Out[70]:
```
{(11, 22, 33), 10, 20, 'harsh'}
```

In [71]:
```
myset4=set()
print(type(myset4))
```

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

In [72]:
```
my_set1=set(('one','two','three','four'))
my_set1
```

Out[72]:
```
{'four', 'one', 'three', 'two'}
```

In [73]:
```
myset = {'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight'}
for i in myset:
    print(i)
```

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

In [74]:
```
for i in enumerate(myset):
    print(i)
```

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

In [75]:
```
myset
```

Out[75]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [76]:
```
myset.add('NINE')
myset
```

Out[76]:
```
{'NINE', 'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [77]:
```
myset.update(['TEN','ELEVEN','TWELVE'])
myset
```

Out[77]:
```
{'ELEVEN',
 'NINE',
 'TEN',
 'TWELVE',
 'eight',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'three',
 'two'}
```

In [78]:
```
myset.remove('NINE')
myset
```

Out[78]:
```
{'ELEVEN',
 'TEN',
 'TWELVE',
 'eight',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'three',
 'two'}
```

In [79]:
```
myset.clear()
myset
```

Out[79]:
```
set()
```

In [80]:
```
del myset
myset
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[80], line 2
      1 del myset
----> 2 myset

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

In [81]:
```
myset = {'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight'}
myset
```

Out[81]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [82]:
```
myset1=myset
myset1
```

Out[82]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [83]:
```
my_set = myset.copy()
my_set
```

Out[83]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [84]:
```
myset.add('nine')
myset
```

Out[84]:
```
{'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}
```

In [85]:
```
myset1
```

Out[85]:
```
{'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}
```

In [86]:
```
my_set#copied element won't change
```

Out[86]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

Union,intersection,disjoint

In [87]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
C = {8,9,10}
```

In [88]:
```
A|B
```

Out[88]:
```
{1, 2, 3, 4, 5, 6, 7, 8}
```

OR

In [89]:
```
A.union(B)
```

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

In [90]:
```
A.union(B,C)
```

Out[90]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

OR

In [91]:
```
A|B|C
```

Out[91]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In [92]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [93]:
```
A&B
```

Out[93]:
```
{4, 5}
```

OR

In [94]:
```
A.intersection(B)
```

Out[94]:
```
{4, 5}
```

In [95]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [96]:
```
A-B#Difference
```

Out[96]:
```
{1, 2, 3}
```

In [97]:
```
B-A
```

Out[97]:
```
{6, 7, 8}
```

In [98]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [99]:
```
A^B#Symetric difference
```

Out[99]:
```
{1, 2, 3, 6, 7, 8}
```

subset

In [100…
```
A = {1,2,3,4,5,6,7,8,9}
B = {3,4,5,6,7,8}
C = {10,20,30,40}
```

In [101…
```
A.issuperset(B)
```
Out[101… True

In [102…
```
B.issuperset(A)
```
Out[102… False

In [103…
```
C.isdisjoint(A)
```
Out[103… True

In [104…
```
B.isdisjoint(A)
```
Out[104… False

dICTIONARY

In [105…
```
mydict = dict()
mydict
```
Out[105… {}

In [106…
```
mydict = {1:'one' , 2:'two' , 3:'three'}
mydict
```
Out[106… {1: 'one', 2: 'two', 3: 'three'}

In [107…
```
mydict = dict({1:'one' , 2:'two' , 3:'three'})
mydict
```
Out[107… {1: 'one', 2: 'two', 3: 'three'}

In [108…
```
mydict = {1:'one' , 'A':'two' , 3:'three'}
mydict
```
Out[108… {1: 'one', 'A': 'two', 3: 'three'}

In [109…
```
mydict.keys()
```

Out[109… dict_keys([1, 'A', 3])

In [110…
```
mydict.values()
```

Out[110… dict_values(['one', 'two', 'three'])

In [111…
```
mydict.items()
```

Out[111… dict_items([(1, 'one'), ('A', 'two'), (3, 'three')])

In [112…
```
mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria']}
mydict
```

Out[112…
```
{1: 'one', 2: 'two', 'A': ['asif', 'john', 'Maria']}
```

In [113…
```
mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria'], 'B':('Bat' , 'cat','hat')}
mydict
```

Out[113…
```
{1: 'one',
 2: 'two',
 'A': ['asif', 'john', 'Maria'],
 'B': ('Bat', 'cat', 'hat')}
```

In [115…
```
mydict = {1:'one' , 2:'two' , 'A':{'Name':'asif' , 'Age':23}, 'B':('Bat' ,'cat','hat')}
mydict
```

Out[115…
```
{1: 'one',
 2: 'two',
 'A': {'Name': 'asif', 'Age': 23},
 'B': ('Bat', 'cat', 'hat')}
```

In [116…
```
keys = {'a' , 'b' , 'c' , 'd'}
mydict3 = dict.fromkeys(keys)
mydict3
```

Out[116…
```
{'b': None, 'a': None, 'd': None, 'c': None}
```

In [117…
```
keys = {'a' , 'b' , 'c' , 'd'}
value=24
mydict3 = dict.fromkeys(keys,value)
mydict3
```

Out[117…
```
{'b': 24, 'a': 24, 'd': 24, 'c': 24}
```

In [118…
```
keys = {'a' , 'b' , 'c' , 'd'}
value=[24,34,44]
mydict3 = dict.fromkeys(keys,value)
mydict3
```

Out[118…
```
{'b': [24, 34, 44], 'a': [24, 34, 44], 'd': [24, 34, 44], 'c': [24, 34, 44]}
```

In [119…
```
value.append(40)
mydict3
```

Out[119…
```
{'b': [24, 34, 44, 40],
 'a': [24, 34, 44, 40],
 'd': [24, 34, 44, 40],
 'c': [24, 34, 44, 40]}
```

In [120…
```
mydict
```

Out[120…
```
{1: 'one',
 2: 'two',
 'A': {'Name': 'asif', 'Age': 23},
 'B': ('Bat', 'cat', 'hat')}
```

In [121…
```
mydict[1]
```

Out[121…
```
'one'
```

In [122…
```
mydict1 = {'Name':'Asif' , 'ID': 74123 , 'DOB': 1991 , 'job' :'Analyst'}
mydict1
```

Out[122…
```
{'Name': 'Asif', 'ID': 74123, 'DOB': 1991, 'job': 'Analyst'}
```

In [123…
```
mydict1['Name']
```

Out[123…
```
'Asif'
```

In [124…
```
mydict1
```

Out[124…
```
{'Name': 'Asif', 'ID': 74123, 'DOB': 1991, 'job': 'Analyst'}
```

In [125…
```
mydict1['DOB']=1996
mydict1['job']='testing'
mydict1
```

Out[125…
```
{'Name': 'Asif', 'ID': 74123, 'DOB': 1996, 'job': 'testing'}
```

In [126…
```
mydict.popitem()
```

Out[126…
```
('B', ('Bat', 'cat', 'hat'))
```

In [127…
```
mydict1.popitem()
```

Out[127…
```
('job', 'testing')
```

In [128…
```
mydict1
```

Out[128…
```
{'Name': 'Asif', 'ID': 74123, 'DOB': 1996}
```

In [129…
```
del[mydict1['ID']]
mydict1
```

Out[129…
```
{'Name': 'Asif', 'DOB': 1996}
```

In [130…
```
mydict1.clear()
mydict1
```

Out[130…
```
{}
```

In [131…
```
del mydict1
mydict1
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[131], line 2
      1 del mydict1
----> 2 mydict1

NameError: name 'mydict1' is not defined
```

In [132…
```
mydict = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Hilsinki'}
mydict
```

Out[132…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsinki'}
```

In [ ]:
```
mydict1=mydict
mydict1
```

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

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

In [ ]:
```
mydict1
```

In [ ]:
```
mydict = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Hilsinki','Job':'testing'}
mydict
```

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

In [ ]:
```
mydict
```

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

In [ ]:
```

```