Tuple Creation

In [11]:
```
tup1 = ()      # Emppty tuple
tup1
```

Out[11]: ()

In [13]:
```
tup2 = (10,30,60)   # tuple of integers number
tup2
```

Out[13]: (10, 30, 60)

In [15]:
```
tup3 = (10,77,30,66,60.89)  # tuple  of float number
tup3
```

Out[15]: (10, 77, 30, 66, 60.89)

In [17]:
```
tup4 = ('one','two',"three")  # tuple of string
tup4
```

Out[17]: ('one', 'two', 'three')

In [19]:
```
tup5 = ('vaishu',25,(50,100),(150,90))  # nested tuple
tup5
```

Out[19]: ('vaishu', 25, (50, 100), (150, 90))

In [21]:
```
tup6 = (100,'vaishu',17.765) # Tuple of mixed data types
tup6
```

Out[21]: (100, 'vaishu', 17.765)

In [25]:
```
tup7 = ('vaishu',25,[50,100],[150,90],{'sapna','sangeeta'},(99,22,33))
tup7
```

Out[25]: ('vaishu', 25, [50, 100], [150, 90], {'sangeeta', 'sapna'}, (99, 22, 33))

In [27]:
```
len(tup7)  # Length of list
```

Out[27]: 6

Tuple Indexing

In [31]:
```
tup2[0] # Retrive first element of the tuple
```

Out[31]: 10

In [33]:
```
tup4[0] # Retreive first element of the tuple
```

Out[33]: 'one'

In [35]:
```
tup4[0][0] # Nested imdexing - Access the first chsrscter of the tuple element
```

Out[35]: 'o'

In [37]:
```
tup4[-1] #Last item of the tuple
```

Out[37]: 'three'

In [39]:
```
tup5[-1] # Last item of the tuple
```

Out[39]: (150, 90)

Tuple Slicing

In [78]:
```
mytuple = ('one','two','three','four','five','six','seven','eight')
mytuple
```

Out[78]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')

In [80]:
```
mytuple[0:3] # Return all item from 0th to 3rd index location excluding the item
```

Out[80]: ('one', 'two', 'three')

In [82]:
```
mytuple[2:5] # list all items from 2nd to 5th index location excluding the item
```

Out[82]: ('three', 'four', 'five')

In [84]:
```
mytuple[:3] # Return first three items
```

Out[84]: ('one', 'two', 'three')

In [86]:
```
mytuple[:2] # Return first two items
```

Out[86]: ('one', 'two')

In [88]:
```
mytuple[-3:] # Return last three items
```

Out[88]: ('six', 'seven', 'eight')

In [90]:
```
mytuple[-2:] # Return last two items
```

Out[90]: ('seven', 'eight')

In [92]:
```
mytuple[-1:] # Return last item of the tuple
```

Out[92]: ('eight',)

In [94]:
```
mytuple[:] # Return whole tuple
```

Out[94]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')

Remove & Change items

In [97]:
```
mytuple
```

Out[97]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')

In [99]:
```
del mytuple[0] # Tuple are immutable which means we can't DELETE tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[99], line 1
----> 1 del mytuple[0]

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

In [101…
```
mytuple[0] = 1 # Tuple are immutable which means we can't CHANGE tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[101], line 1
----> 1 mytuple[0] = 1

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

In [103…
```
del mytuple # Delete entire tuple object is possible
```

LOOP through a tuple

In [108…
```
mytuple = ('one','two','three','four','five','six','seven','eight')
mytuple
```

Out[108… ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')

In [110…
```
for i in mytuple:
    print(i)
```

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

In [112…
```
for i in enumerate(mytuple):
    print(i)
```

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

Tuple Membership

In [125…
```
mytuple
```

Out[125…
```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

In [127…
```
'one' in mytuple #check if 'one' exist in the list
```

Out[127…
```
True
```

In [129…
```
'ten' in mytuple # check if 'ten' exist in the list
```

Out[129…
```
False
```

In [135…
```
if 'three' in mytuple: # check if 'three' exist in the list
    print('Three is present in the tuple')
else:
    print('Three is not present in the tuple')
```

```
Three is present in the tuple
```

In [137…
```
if 'eleven' in mytuple: # check if 'eleven; exist in the list
    print('eleven is present in the tuple')
else:
    print('eleven is not present in the tuple')
```

```
eleven is not present in the tuple
```

Index Position

In [140…
```
mytuple
```

Out[140…
```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

In [144…
```
mytuple.index('one')  # Index of first element equal to 'one'
```

Out[144…
```
0
```

In [146…
```
mytuple.index('five') # Index of first element equal to 'five'
```

Out[146… 4

In [156…
```
mytuple1 = ('one','two','three','four','one','one','two','three')
mytuple1
```

Out[156… ('one', 'two', 'three', 'four', 'one', 'one', 'two', 'three')

In [158…
```
mytuple.index('one') # index of first element equal to 'one'
```

Out[158… 0

Sorting

In [164…
```
mytuple2 = (43,67,99,12,6,90,67)
```

In [166…
```
sorted(mytuple2) # return a new sorted list and doesn't change original tuple
```

Out[166… [6, 12, 43, 67, 67, 90, 99]

In [168…
```
sorted(mytuple2, reverse =True ) # Sort in descending order
```

Out[168… [99, 90, 67, 67, 43, 12, 6]

Count function

In [171…
```
mytuple2
```

Out[171… (43, 67, 99, 12, 6, 90, 67)

In [180…
```
mytuple2.count(67)  # count the  total element of 67
```

Out[180… 2

In [182…
```
mytuple.count(43) # count the total element of 43
```

Out[182… 0

In [ ]: