Tuple Creation

In [1]:
```
tup1 =()       # empty tuple
```

In [2]:
```
tup2 = (10,30,60)  #tuple of integers numbers
```

In [3]:
```
tup3 = (10.77,30.66,60.89) # tuple of float numbers
```

In [4]:
```
tup4 = ('one','two',"three") # tuple of strings
```

In [5]:
```
tup5 =('Asif', 25,(50,100),(130,69)) # Nested tuples
```

In [6]:
```
tup6 =(100,'Asif',17.65) # Tuple of mixed data types
```

In [7]:
```
tup7 = ('Asif',25,[30.100],[130,89],{'John','abhay'},(99,22,50))
```

In [8]:
```
len(tup7) #lenth of list1
```

Out[8]:
```
6
```

Tuple indexing

In [9]:
```
tup2[0]    #retrive first element of the tuple
```

Out[9]:
```
10
```

In [10]:
```
tup4[2]    #retrive first element of the tuple
```

Out[10]:
```
'three'
```

In [11]:
```
tup4[0][0] # Nested indexing - Access the first character of the first tuple e
```

Out[11]:
```
'o'
```

In [12]:
```
tup4[-1]
```

Out[12]:
```
'three'
```

In [13]:
```
tup5[-1]
```

Out[13]:
```
(130, 69)
```

Tuple slicing

In [14]:
```
mytuple = ('one','two','three','four','five','six','seven','eight')
```

In [15]:
```
mytuple[0:4] # return all items from 0 th to 3rd  index location excluding the
```

Out[15]:
```
('one', 'two', 'three', 'four')
```

In [16]:
```
mytuple[2:5]   # return all items from 2 th to 5 th index location excluding t
```

Out[16]:
```
('three', 'four', 'five')
```

In [17]:
```
mytuple[:6]   # Return first 6 items
```

Out[17]:
```
('one', 'two', 'three', 'four', 'five', 'six')
```

In [18]:
```
mytuple[:3]   # Return first 3 items
```

Out[18]:
```
('one', 'two', 'three')
```

In [19]:
```
mytuple[-3:]    # Return last 3 items
```

Out[19]:
```
('six', 'seven', 'eight')
```

In [20]:
```
mytuple[-5:] # Return last 5 items
```

Out[20]:
```
('four', 'five', 'six', 'seven', 'eight')
```

In [21]:
```
mytuple[-1] # return last item of the tuple
```

Out[21]:
```
'eight'
```

In [22]:
```
mytuple[:]  # return whole tuple
```

Out[22]:
```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

Remove & Change Items

In [23]:
```
mytuple
```

Out[23]:
```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

In [24]:
```
del mytuple[0]  # Tuples are immutable which means we can't DELETE tuple items
```

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

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

In [25]:
```
mytuple[0] = 1  # Tuples are immutable which means we can't CHANGE tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[25], line 1
----> 1 mytuple[0] = 1

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

In [26]:
```
del mytuple # Deleting entire tuple object is possible
```

Loop through a tuple

In [27]:
```
mytuple = ('one','two','three','four','five','six','seven','eight')
```

In [28]:
```
for i in mytuple:
    print(i)
```

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

In [29]:
```
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 [30]:
```
mytuple
```

Out[30]:
```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

In [31]:
```
'one' in mytuple # check if 'one' exist in the list
```

Out[31]:
```
True
```

In [32]:
```
'ten' in mytuple # check if 'ten' exist in the list
```

Out[32]:
```
False
```

In [33]:
```
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 [34]:
```
if 'eleven' in mytuple:          # Check if 'three' 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 [35]:
```
mytuple
```

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

In [36]:
```
mytuple.index('one') # Index of first element equal to 'one'
```

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

In [37]:
```
mytuple.index('two') # Index of first element equal to 'one'
```

Out[37]:
```
1
```

In [39]:
```
mytuple1 = ('one','two','three','four','five','six','seven')
```

In [40]:
```
mytuple1.index('one')
```

Out[40]:
```
0
```

Sorting

In [41]:
```
mytuple2 =(43,45,43,23,45,76,56,13)
```

In [42]:
```
sorted(mytuple2)  # Returns a new sorted list and doen't change original tuple
```

Out[42]:
```
[13, 23, 43, 43, 45, 45, 56, 76]
```

In [44]:
```
sorted(mytuple2, reverse=True) # Sort in descending order
```

Out[44]:
```
[76, 56, 45, 45, 43, 43, 23, 13]
```