Tuples

"""

1. Tuple is similar to List except that the objects in tuple are immutable which means we cannot change the elements of a tuple once assigned.
2. When we do not want to change the data over time, tuple is a preferred data type.
3. Iterating over the elements of a tuple is faster compared to iterating over a list. """

In [3]:
```
tup1 = () # Empty tuple
```

In [4]:
```
tup2 = (10, 20, 30) # tuple of integers numbers
```

In [5]:
```
tup3 = (10.77,30.66,60.89) # tuple of float numbers
```

In [6]:
```
tup4 = ('one','two' , "three") # tuple of strings
```

In [7]:
```
tup5 = ('Asif', 25 ,(50, 100),(150, 90)) # Nested tuples
```

In [8]:
```
tup6 = (100, 'Asif', 17.765) # Tuple of mixed data types
```

In [9]:
```
tup7 = ('Asif', 25 ,[50, 100],[150, 90] , {'John' , 'David'} , (99,22,33)) # Tuple
```

In [10]:
```
len(tup7) #Length of list
```

Out[10]:
```
6
```

Tuple Indexing

In [11]:
```
tup2[0] # Retreive first element of the tuple
```

Out[11]:
```
10
```

In [12]:
```
tup4[0] # Retreive first element of the tuple
```

Out[12]:
```
'one'
```

In [13]:
```
tup4[0][0] # Nested indexing - Access the first character of the first tuple elemen
```

Out[13]:
```
'o'
```

In [14]:
```
tup4[-1] # Last item of the tuple
```

Out[14]:
```
'three'
```

```
In [15]: tup5[-1] # Last item of the tuple
```

```
Out[15]: (150, 90)
```

Tuple Slicing

```
In [16]: mytuple = ('one' , 'two' , 'three' , 'four' , 'five' , 'six' , 'seven' , 'eight')
```

```
In [17]: mytuple[0:3] # Return all items from 0th to 3rd index location excluding the item a
```

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

```
In [18]: mytuple
```

```
Out[18]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [19]: mytuple[2:5] # List all items from 2nd to 5th index location excluding the item at
```

```
Out[19]: ('three', 'four', 'five')
```

```
In [20]: mytuple [:3]  # Return first three items
```

```
Out[20]: ('one', 'two', 'three')
```

```
In [21]: mytuple[:2] # Return first two items
```

```
Out[21]: ('one', 'two')
```

```
In [23]: mytuple [-3] # Return last third item
```

```
Out[23]: 'six'
```

```
In [24]: mytuple [-2:] # Return last two items
```

```
Out[24]: ('seven', 'eight')
```

```
In [25]: mytuple[-1] # Return last item of the tuple
```

```
Out[25]: 'eight'
```

```
In [26]: mytuple[:] # Return whole tuple
```

```
Out[26]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

Remove & Change Items

```
In [27]: mytuple
```

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

```
In [28]: del mytuple[0] # Tuples are immutable which means we can't DELETE tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[28], line 1
----> 1 del mytuple[0]

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

```
In [31]: mytuple [0] = 1 # Tuples are immutable which means we cants change tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[31], line 1
----> 1 mytuple [0] = 1

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

```
In [32]: del mytuple # Deleting entire tuple object is possible
```

```
In [33]: mytuple
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[33], line 1
----> 1 mytuple

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

Loop through a tuple

```
In [34]: mytuple
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[34], line 1
----> 1 mytuple

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

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

```
In [36]: mytuple
```

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

```
In [37]: for i in mytuple:
             print (i)
```

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

In [39]:

```
for i in enumerate(mytuple):
    print(i)
```

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

Count

In [40]:

```
mytuple1 =('one', 'two', 'three', 'four', 'one', 'one', 'two', 'three')
```

In [41]:

```
mytuple1.count('one') # Number of times item "one" occurred in the tuple.
```

Out[41]:

```
3
```

In [42]:

```
mytuple1.count ('two') #Occurance of item 'two' in the tuple
```

Out[42]:

```
2
```

In [43]:

```
mytuple1.count('two') # Occurence of item 'two' in the tuple
```

Out[43]:

```
2
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: