LIST CREATION

In [88]:
```
list1 = [] # Empty List
```

In [2]:
```
print(type(list1))
```

```
<class 'list'>
```

In [3]:
```
list2 = [10,30,60] # List of integers numbers
```

In [4]:
```
list3 = [10.77,30.66,60.89]    # List of float numbers
```

In [14]:
```
list4 = ['one','two',"three"]   # List of strings
```

In [6]:
```
list5 = ['Sidra', 25, [50,100],[150,90]]  # Nested Lists
```

In [7]:
```
list6 = [100, 'Sidra', 17.765]  # List of mixed daa types
```

In [8]:
```
list7 = ['Sidra', 25, [50,10], [150,90], {'john','David'}]
```

In [9]:
```
len(list6)  # Length of list
```

Out[9]:
```
3
```

LIST INDEXING

In [10]:
```
list2[0]  # Retrieve first element of the list
```

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

In [11]:
```
list4[0]  # Retrieve first element of the list
```

Out[11]:
```
'one'
```

In [12]:
```
list4[0][0]  # Nested indexing - Access the first character of the first list element
```

Out[12]:
```
'o'
```

In [15]:
```
list4[-1]  # Last item of the list
```

Out[15]:
```
'three'
```

In [17]:
```
list5[-1] # Last item of the list
```

Out[17]:
```
[150, 90]
```

LIST SLICING

In [18]:
```
mylist = ['one', 'two', 'three', 'four', 'five','six', 'seven', 'eight']
```

In [19]:
```
mylist[0:3]  # Return all item from 0th to 3rd index location excluding the item
```

Out[19]:
```
['one', 'two', 'three']
```

In [20]:
```
mylist[2:5]  # List all item from 2nd to 5th index location excluding the item
```

Out[20]:
```
['three', 'four', 'five']
```

In [21]:
```
mylist[:3]  # Return first three items
```

Out[21]:
```
['one', 'two', 'three']
```

In [22]:
```
mylist[:2]  # Return first two items
```

Out[22]:
```
['one', 'two']
```

In [23]:
```
mylist[-3:]  # Return last three items
```

Out[23]:
```
['six', 'seven', 'eight']
```

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

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

In [25]:
```
mylist[-1]  # Return last items of the list
```

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

In [26]:
```
mylist[:]  # Return whole list
```

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

ADD, REMOVE, & CHANGE ITEMS

In [32]:
```
mylist
```

Out[32]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'nine',
 'ten',
 'nine',
 'nine']
```

In [33]:
```
mylist.append('nine')  # Add an item to the end of the list
mylist
```

Out[33]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'nine',
 'ten',
 'nine',
 'nine',
 'nine']
```

In [34]:
```
mylist.insert(9,'ten')  # Add item at index location 9
mylist
```

Out[34]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'nine',
 'ten',
 'ten',
 'nine',
 'nine',
 'nine']
```

In [35]:
```
mylist.insert(1,'ONE')  # Add item at index location 1
mylist
```

Out[35]:
```
['one',
 'ONE',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'nine',
 'ten',
 'ten',
 'nine',
 'nine',
 'nine']
```

In [36]:
```
mylist.remove('ONE')  # Remove item "ONE"
mylist
```

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

In [37]:
```
mylist.pop() # Remove last item of the list
mylist
```

Out[37]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'nine',
 'ten',
 'ten',
 'nine',
 'nine']
```

In [38]:
```
mylist.pop(8) # Remove item at index location 8
mylist
```

Out[38]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'eight',
 'ten',
 'ten',
 'nine',
 'nine']
```

In [39]:
```
del mylist[7] # remove item at index location 7
mylist
```

Out[39]:
```
['one',
 'two',
 'three',
 'four',
 'five',
 'six',
 'seven',
 'ten',
 'ten',
 'nine',
 'nine']
```

In [40]:
```
# change value of the string
mylist[0] = 1
mylist[1] = 2
mylist[2] = 3
mylist
```

Out[40]:
```
[1, 2, 3, 'four', 'five', 'six', 'seven', 'ten', 'ten', 'nine', 'nine']
```

In [41]:
```
mylist.clear() # Empty list / Delete all items in the list
mylist
```

Out[41]:
```
[]
```

In [42]:
```
del mylist # Delete the whole list
mylist
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [42], in <cell line: 2>()
      1 del mylist # Delete the whole list
----> 2 mylist

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

COPY LIST

In [43]:
```
mylist = ['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine']
```

In [44]:
```
mylist1 = mylist # Create a new reference 'mylist1'
```

In [45]:
```
id(mylist) , id(mylist1) # The address of both mylist and mylist1 will be the same
```

Out[45]:
```
(3043404859072, 3043404859072)
```

In [46]:
```
mylist2 = mylist.copy() # Create a copy of the list
```

In [47]:
```
id(mylist2) # The address of mylist2 will be different from mylist because mylist2 is cop
```

Out[47]:
```
3043404858752
```

In [48]:
```
mylist[0] = 1
```

In [49]:
```
mylist
```

Out[49]:
```
[1, 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine']
```

In [50]:
```
mylist1 # mylist1 willlbe also impacted as it is pointing to the same list
```

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

In [51]:
```
mylist2 # copy of list won't be impacted due to changes made on the original list
```

Out[51]:
```
['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine']
```

JOIN LISTS

In [53]:
```
list1 = ['one', 'two', 'three', 'four']
list2 = ['five', 'six', 'seven', 'eight']
```

In [54]:
```
list3 = list1 + list2 # Join two lists by '+' operator
list3
```

Out[54]:
```
['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight']
```

In [55]:
```
list1.extend(list2) # Append list2 with list1
list1
```

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

LIST MEMBERSHIP

In [56]:
```
list1
```

Out[56]:
```
['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight']
```

In [57]:
```
'one' in list1 # Check if 'one' exist in the list
```

Out[57]:
```
True
```

In [58]:
```
'ten' in list1 # Check if 'ten' exist in the list
```

Out[58]:
```
False
```

In [59]:
```
if 'three' in list1: # Check if 'three' exist in the list
    print('Three is present in the list')
else:
    print('Three is not present in the list')
```

```
Three is present in the list
```

In [60]:
```
if 'eleven' in list1: # Check if 'eleven' exist in the list
    print('eleven is present in the list')
else:
    print('eleven is not present in the list')
```

```
eleven is not present in the list
```

REVERSE & SORT LIST

In [61]:
```
list1
```

Out[61]:
```
['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight']
```

In [63]:
```
list1.reverse() # Reverse the list
list1
```

Out[63]:
```
['eight', 'seven', 'six', 'five', 'four', 'three', 'two', 'one']
```

In [64]:
```
list1 = list1[::-1] # Reverse the list
list1
```

Out[64]:
```
['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight']
```

In [65]:
```
mylist3 = [9,5,2,99,12,88,34]
mylist3.sort()  # Sort list in ascending order
mylist3
```

Out[65]: [2, 5, 9, 12, 34, 88, 99]

In [66]:
```
mylist = [9,5,2,99,12,88,34]
mylist3.sort(reverse=True) # Sort list in decending order
mylist3
```

Out[66]: [99, 88, 34, 12, 9, 5, 2]

In [67]:
```
mylist4 = [88,65,33,21,11,98]
sorted(mylist4)   # Returns a new sorted list and doesn't change original list
```

Out[67]: [11, 21, 33, 65, 88, 98]

In [68]:
```
mylist4
```

Out[68]: [88, 65, 33, 21, 11, 98]

LOOP THROUGH A LIST

In [69]:
```
list1
```

Out[69]: ['one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight']

In [70]:
```
for i in list1:
    print(i)
```

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

COUNT

In [71]:
```
list10 = ['one', 'two', 'three', 'four', 'one', 'one', 'two', 'three']
```

In [76]:
```
list10.count('one') # Number of item "one" occured in the list
```

Out[76]: 3

In [73]:
```
list10.count('two') # Occurence of "two" in the list
```

Out[73]: 2

In [74]:
```
list10.count('four') # Occurence of "four" in the list
```

Out[74]: 1

ALL / ANY

In [77]:
```
# The all() methods returns:

# .True- If all elements in a list are true
# .False- If any element in a list is false

# The any() function returns True if any element in the list is True.If not,any() returns
```

In [78]:
```
L1 = [1,2,3,4,0]
```

In [79]:
```
all(L1) # Will return false as one value is false (Value 0)
```

Out[79]:
```
False
```

In [80]:
```
any(L1) # Will return True as we have items in the list with True value
```

Out[80]:
```
True
```

In [81]:
```
L2 = [1,2,3,4,True,False]
```

In [83]:
```
all(L2) # Return false as one value is false
```

Out[83]:
```
False
```

In [84]:
```
any(L2) # Will return True as we  have items in the list with True value
```

Out[84]:
```
True
```

In [85]:
```
L3 = [1,2,3,True]
```

In [86]:
```
all(L3) # Will return True as all items in the list are True
```

Out[86]:
```
True
```