In [1]:

```
'''
The notebook gives examples on following methods of list

append() - Add an element to the end of the list
extend() - Add all elements of a list to the another list
insert()- I nsert an item at the defined index
remove() - Removes an item from the list
pop() - Removes and returns an element at the given index
clear() - Removes all items from the list
index() - Returns the index of the first matched item
count() - Returns the count of number of items passed as an argument
sort() - Sort items in a list in ascending order
reverse() - Reverse the order of items in the list
copy() - Returns a shallow copy of the list
Built-in Functions with List
all() Return True if all elements of the list are true
        (or if the list is empty).
any() Return True if any element of the list is true.
        If the list is empty, return False.
enumerate() Return an enumerate object. It contains the index
            and value of all the items of list as a tuple.
len() Return the length (the number of items) in the list.
list() Convert an iterable (tuple, string, set, dictionary) to a list.
max() Return the largest item in the list.
min() Return the smallest item in the list
sorted() Return a new sorted list (does not sort the list itself).
sum() Return the sum of all elements in the list.
'''
pass
```

In [2]:

```
#creating list
my_shopping_list = ['apple', 'banana', 'mango']
print(my_shopping_list)
```

```
['apple', 'banana', 'mango']
```

In [3]:

```
#adding an element at end
my_shopping_list.append('pineapple')
print(my_shopping_list)
```

```
['apple', 'banana', 'mango', 'pineapple']
```

In [4]:

```
#adding multiple element
my_shopping_list.extend(['table', 'chair', 'fan'])
print(my_shopping_list)
```

```
['apple', 'banana', 'mango', 'pineapple', 'table', 'chair', 'fan']
```

In [5]:

```
#adding element at perticular location
my_shopping_list.insert(1, 'cake')
print(my_shopping_list)
```

```
['apple', 'cake', 'banana', 'mango', 'pineapple', 'table', 'chair', 'fan']
```

In [6]:

```
#remove an elemenet
my_shopping_list.remove('chair')
print(my_shopping_list)
```

```
['apple', 'cake', 'banana', 'mango', 'pineapple', 'table', 'fan']
```

In [7]:

```
#remove an element and return it
my_fourth_element = my_shopping_list.pop(4)
print(my_shopping_list)
print(my_fourth_element)
```

```
['apple', 'cake', 'banana', 'mango', 'table', 'fan']
pineapple
```

In [8]:

```
#make the list empty
my_shopping_list.clear()
print(my_shopping_list)
```

```
[]
```

In [9]:

```
#deducing index of an element
my_shopping_list = ['apple', 'cake', 'banana', 'mango', 'pineapple', 'table', 'chair'
index_of_mango = my_shopping_list.index('mango')
print(index_of_mango)
```

```
3
```

In [10]:

```
#counting an element
num_cake = my_shopping_list.count('cake')
print(num_cake)
```

```
2
```

In [11]:

```
#sorting the list
my_shopping_list.sort()
print(my_shopping_list)
```

```
['apple', 'banana', 'cake', 'cake', 'chair', 'fan', 'mango', 'pineapple', 'table']
```

In [12]:

```
#reversing the list
my_shopping_list.reverse()
print(my_shopping_list)
```

```
['table', 'pineapple', 'mango', 'fan', 'chair', 'cake', 'cake', 'banana', 'apple']
```

In [13]:

```
#boolean operation
my_bollean_list = [True, False, True]
all(my_bollean_list)
```

Out[13]:

```
False
```

In [14]:

```
#boolean operation
any(my_bollean_list)
```

Out[14]:

```
True
```

In [15]:

```
#enumerating the list
for index, name in enumerate(my_shopping_list):
    print(f'element at {index}: {name}')
```

```
element at 0: table
element at 1: pineapple
element at 2: mango
element at 3: fan
element at 4: chair
element at 5: cake
element at 6: cake
element at 7: banana
element at 8: apple
```

In [16]:

```
#length of the list
len_of_shopping = len(my_shopping_list)
print(f'number of elemenets in shopping cart is {len_of_shopping}')
```

```
number of elemenets in shopping cart is 9
```

In [17]:

```
#converting to the list
my_tuple = ('Gaurav', 'Kumar')
my_list = list(my_tuple)
print(my_list)
print(type(my_list))
```

```
['Gaurav', 'Kumar']
<class 'list'>
```

In [18]:

```
#max, min, sort, sum a list
my_numerical_list = [9,7,4,2,3,4,1,2,0,1]
print(f'numerical list: {my_numerical_list}')
print(f'max number: {max(my_numerical_list)}')
print(f'min number: {min(my_numerical_list)}')
print(f'sorted numbers: {sorted(my_numerical_list)}')
print(f'sum of numbers: {sum(my_numerical_list)}')
```

```
numerical list: [9, 7, 4, 2, 3, 4, 1, 2, 0, 1]
max number: 9
min number: 0
sorted numbers: [0, 1, 1, 2, 2, 3, 4, 4, 7, 9]
sum of numbers: 33
```