In this file we practice some of the useful methods and tricks for lists. Let's get started!

As dicsussed in class, lists are ordered, mutable and allow duplicate values:

In [1]:
```
ex = [1,2,3,[2,5,7], "hey", "you", 3]
```

In [2]:
```
ex[0]
```

Out[2]:
```
1
```

In [3]:
```
ex[0] = 7
print(ex[0])
```

```
7
```

1-The first method is append(). This method adds an item to the end of the list:

In [4]:
```
ex.append(25)
print(ex)
```

```
[7, 2, 3, [2, 5, 7], 'hey', 'you', 3, 25]
```

As you can see, 25 was added to the end of the list.

2-The second method is insert(). You can also add an item to the list using this method. But with this method you can choose the index to add the item to:

In [5]:
```
ex.insert(2,"item was inserted here")
print(ex)
```

```
[7, 2, 'item was inserted here', 3, [2, 5, 7], 'hey', 'you', 3, 25]
```

In [6]:
```
ex[4].insert(0,"new item inserted")
print(ex[4])
```

```
['new item inserted', 2, 5, 7]
```

As you can see the item was inserted at index 2

Now that we've learned about adding items to lists, let's find out how to remove them.

3- The third method is the remove() method. As you can probably guess this method removes an item from the list. This method is used when you want to remove an item from a list by it's value.

In [7]:
```
ex.remove(2)
print(ex)
```

```
[7, 'item was inserted here', 3, ['new item inserted', 2, 5, 7], 'hey', 'you', 3, 25]
```

as you can see 2 was removed from the list.

4-If you want to remove an item by it's index from a list you can use the pop() method.

In [21]:
```
ex = ["hey", "you", "out", "there"]
ex.pop(1)
```

Out[21]:
```
'you'
```

In [22]:
```
print(ex)
```

```
['hey', 'out', 'there']
```

pop() method also returns the value of the item that has been removed:

In [9]:
```
a = ex.pop(0)
print(a)
```

```
hey
```

5- If you ever need to add items of a list to another list, you can use one of the following ways:

In [23]:
```
list1 = [1,2,3,4,5,6]
list2 = [6,7,8,9,10]
list1.extend(list2)
print(list1)
```

```
[1, 2, 3, 4, 5, 6, 6, 7, 8, 9, 10]
```

In [11]:
```
list1 = [1,2,3,4,5]
list2 = [6,7,8,9,10]
list1 = list1 + list2
print(list1)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

In [25]:
```
list1= [1,2,3,4]
list2 = list1
print(list2)
```

```
[1, 2, 3, 4]
```

In [26]:
```
list2.append(5)
```

In [27]:
```
print(list2)
```

```
[1, 2, 3, 4, 5]
```

In [28]:
```
print(list1)
```

```
[1, 2, 3, 4, 5]
```

6-And if you need to copy a list, you can use the "copy()" method:

In [12]:
```
list1 = ["What's", "in", "the", "boxxxxxxxx?"]
list2 = list1.copy()
print(list2)
```

```
["What's", 'in', 'the', 'boxxxxxxxx?']
```

In [ ]:
```

```

As discussed in the class, if you need to copy a list, you should use the copy() method to avoid shallow copy problem.

7- Now, Let's discuss sorting lists.For sorting lists you can use the "sort()" method:

In [13]:
```
ex = [-1.1,-3,6,4,7,11,167]
ex.sort()
print(ex)
```

```
[-3, -1.1, 4, 6, 7, 11, 167]
```

You can also use the built-in funcion "sorted()". Unlike the previous method, this function doesn't change the original list, but returns a sorted list:

In [30]:
```
ex = [-1.1,-3,6,4,7,11,167]
sorted_ex = sorted(ex)
print(sorted_ex)
```

```
[-3, -1.1, 4, 6, 7, 11, 167]
```

In [31]:
```
print(ex)
```

```
[-1.1, -3, 6, 4, 7, 11, 167]
```

8- For clearing a list, You can use one of the following ways:

In [15]:
```
ex = [1,2,3,4,5,6]
```

In [16]:
```
ex.clear() #removes all the items, in the list
```

In [17]:
```
ex = []
```

9-Sometimes we need to count the number of times that an item has appeared in the list:

In [18]:
```
ex = [1,2,2,2,3,1,"amir", "reza", "amir"]
print(ex.count(2))
print(ex.count("amir"))
```

```
3
2
```

10-And for the last method, if you ever need to reverse the order of items in a list, you can use the "reverse()" method:

In [19]:
```
ex = [1,2,3,4,5,6]
ex.reverse()
print(ex)
```

```
[6, 5, 4, 3, 2, 1]
```

Keep on learning!

In [ ]:
```

```