★ List in Python.

- List is an ordered, mutable, heterogenous collection of python objects.

- Ordered: The sequence of items is preserved.
- Mutable: Updates the existing object (changeable).
- Heterogenous: can hold mutable, immutable objects.

Lets start of with creation of list.

A list is denoted using [] (square brackets)

Example:

```
my_list = [10, 20, 30]  // using square brackets
my_list = list ((10,20,30))  // using constructor.
empty_list = []  // empty list.
```

★ Accessing elements in a list.

As list is a sequence, we can access list items using indexing.

```
my_list[0] // 10
my_list[1] // 20.
```

Just like tuples, we can perform slicing on list.

```
print (my_list[0:2]) // [10, 20]
```

- Mutability.

As list is a mutable object - it can be modified.

```
my_list[0] = 50
print (my_list) // [50, 20, 30]
```

- As list is a mutable object, it comes with various methods.

1. Adding elements.

- appends any object at the end of the list.
  Method - append (object).

2. insert (index, object)

inserts an element at specified index position.

3. extend ([multiple objects])

adds multiple objects to a list.

- Removing Elements.

1. remove (object)

   Removes the first occurence of an element from the list.

2. pop ()

   Removes last item from the list.
   pop (index) - Removes an item present on specified index.

3. clear ()

   Removes all object from the list and returns an empty list.

- Searching / checking.

1. index (object)

   Returns the position (index) of specified object.

2. count (object).

   Returns the count of occurence of specified object in a list.

- Sorting and Reversing.

1. Sort()

   Sorts the list in place.

2. Reverse

   Reverses the stringx list in place.