Table of Contents

Chapter 4

Python Collections (Arrays)

There are four collection data types in the Python programming language:

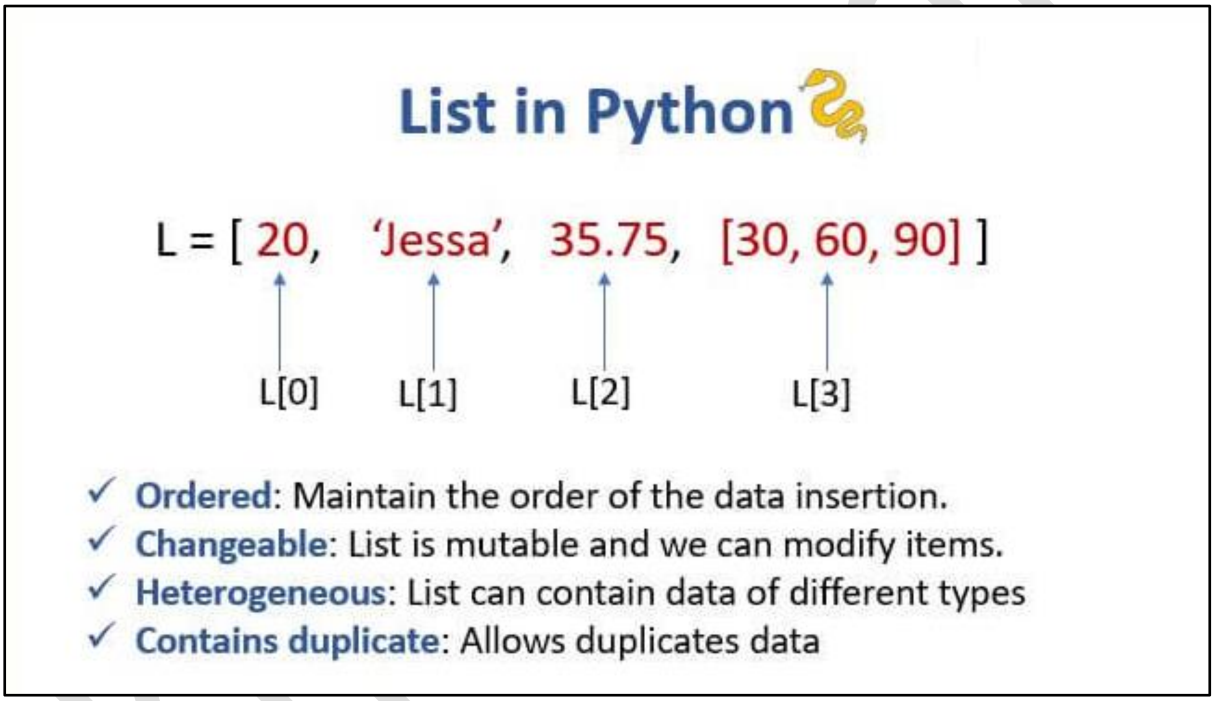
-List is a collection which is ordered and changeable. Allows duplicate members.

-Tuple is a collection which is ordered and unchangeable. Allows duplicate members.

-Set is a collection which is unordered, unchangeable*, and unindexed. No duplicate members.

-Dictionary is a collection which is ordered** and changeable. No duplicate members.

List



List in Python

L = [20, 'Jessa', 35.75, [30, 60, 90]]

✓ **Ordered**: Maintain the order of the data insertion.
✓ **Changeable**: List is mutable and we can modify items.
✓ **Heterogeneous**: List can contain data of different types
✓ **Contains duplicate**: Allows duplicates data

• Lists are used to store multiple items in a single variable.

• Lists are one of 4 built-in data types in Python used to store collections of data, the other 3 are Tuple, Set, and Dictionary, all with different qualities and usage.

• Lists are created using square brackets. List can store value of any data type.

• List items are ordered, changeable, and allow duplicate values.

• List items are indexed, the first item has index [0], the second item has index [1] etc.

• When we say that lists are ordered, it means that the items have a defined order, and that order will not change.

• If you add new items to a list, the new items will be placed at the end of the list.

• The list is changeable, meaning that we can change, add, and remove items in a list after it has been created.

• Since lists are indexed, lists can have items with the same value

Ex.

```
myList = ["Madi","Oily",5,7.5]
print(myList) #["Madi","Oily",5,7.5]
```

Operation on List

```
#Changing Item with index
fruit = ["apple", "banana", "cherry"]
fruit[1] = "blackcurrant"
print(fruit) # ['apple', 'blackcurrant', 'cherry']
#Slicing in List
fruit1 = ["apple", "banana", "cherry", "orange", "kiwi", "mango"]
fruit1[1:3] = ["blackcurrant", "watermelon"]
print(fruit1) # ['apple', 'blackcurrant', 'watermelon', 'orange', 'kiwi', 'mango']
fruit2 = ["apple", "banana", "cherry"]
fruit2[1:2] = ["blackcurrant", "watermelon"]
print(fruit2) # ['apple', 'blackcurrant', 'watermelon', 'cherry']

fruit3 = ["apple", "banana", "cherry"]
fruit3[1:3] = ["watermelon"]
print(fruit3) # ['apple', 'watermelon']

#Add Items
# *********Add Items*********
print("*********Add Items*********")
# Insert Items
# To insert a new list item, without replacing any of the existing values, we can use the insert() method.
# The insert() method inserts an item at the specified index

fruitIns = ["apple", "banana", "cherry"]
fruitIns.insert(2, "watermelon")
print(fruitIns) # ['apple', 'banana', 'watermelon', 'cherry']

# Append
# To add an item to the end of the list, use the append() method
fruitApp = ["apple", "banana", "cherry"]
fruitApp.append("orange")
print(fruitApp) # ['apple', 'banana', 'cherry', 'orange']
```

```
# Extend
# To append elements from another list to the current list, use the extend() method.
# The elements will be added to the end of the list.
fruits = ["apple", "banana", "cherry"]
fruitsExtend = ("mango", "pineapple", "papaya")
fruits.extend(fruitsExtend)
print(fruits) # ['apple', 'banana', 'cherry', 'mango', 'pineapple', 'papaya']

# Remove Item
# *********Remove Item*********
print("*********Remove Item*********")
# remove(item)
# The remove() method removes the specified item.
fruitRem = ["apple", "banana", "cherry"]
fruitRem.remove("banana")
print(fruitRem) # ["apple", "cherry"]

# pop(index)
# The pop() method removes the specified index
# If you do not specify the index, the pop() method removes the last item.
fruitPop = ["apple", "banana", "cherry"]
fruitPop.pop(1) # ["apple", "cherry"]
# fruitPop.pop() #  ["apple", "banana"]
print(fruitPop)

# del listName[index]
fruitDel = ["apple", "banana", "cherry"]
del fruitDel[0] # ['banana', 'cherry']
print(fruitDel)
# The del keyword can also delete the list completely.
del fruitDel
# print(fruitDel) error (fruitDel not defined)

# clear()
# The clear() method empties the list.
# The list still remains, but it has no content.
# Clear the list content:
```

```
fruitClr = ["apple", "banana", "cherry"]
fruitClr.clear()
print(fruitClr)

# Sort item
# *********Sort List*********
print("*********Sort List*********")
# Sort the list alphabetically:
fruitSrt = ["orange", "mango", "kiwi", "pineapple", "banana"]
fruitSrt.sort()
print(fruitSrt) # ['banana', 'kiwi', 'mango', 'orange', 'pineapple']

# Sort the list numerically:
fruitPriceSrt = [100, 50, 65, 82, 23]
fruitPriceSrt.sort()
print(fruitPriceSrt) # [23, 50, 65, 82, 100]

# Sort the list descending:
revSort = [100, 50, 65, 82, 23]
revSort.sort(reverse = True)
print(revSort) # [100, 82, 65, 50, 23]
# or
# revSort.reverse()
# print(revSort)
```

List Methods:

Method	Description
append()	Adds an element at the end of the list
clear()	Removes all the elements from the list
copy()	Returns a copy of the list
count()	Returns the number of elements with the specified value
extend()	Add the elements of a list (or any iterable), to the end of the current list
index()	Returns the index of the first element with the specified value
insert()	Adds an element at the specified position
pop()	Removes the element at the specified position
remove()	Removes the item with the specified value
reverse()	Reverses the order of the list
sort()	Sorts the list