Concatenating Lists using +

We can now create a new List by adding two Lists together.

- a = [1, 2, 3]
  b = [4, 5, 6]
  c = a + b
  print(c)
  [1, 2, 3, 4, 5, 6]
    - print(a)
      [1, 2, 3]

Lists can be sliced using :

t = [9, 41, 12, 3, 74, 15]

t[1:3]
[41, 12]

t[:4]
[9, 41, 12, 3]

t[3:]
[3, 74, 15]

t[:]
[9, 41, 12, 3, 74, 15]

Remember, Just like in strings, the second number is, “up to but not including”

List Methods

x = list()

type(x)
<class ‘list’>

dir(x)
[‘append’, ‘count’, ‘extend’, ‘index’, ‘insert’, ‘pop’, ‘remove’, ‘reverse’, ‘sort’]

https://docs.python.org/tutorial/datastructures.html

Building a List from Scratch

We can create an empty List and then add elements using the append() method.

- stuff = list()
  stuff.append(‘book’)
  stuff.append(99)
  print(stuff)
  [‘book’, 99]

The List stays in order and new elements are added at the end of the List.

- stuff.append(‘cookie’)
  print(stuff)
  [‘book’, 99, ‘cookie’]

Is something in a List

Python provides two operators that let you check if an item is in a List.

- some = [1, 9, 21, 10, 16]
    - 9 in some
      True

These are logical operators that return True or False.

- 15 in some
  False

They do not modify the List.

- 20 not int some
  True

Lists are in Order

A List can hold many items and keep those items in order until we do something to change the order.

- friends = [‘Joseph’, ‘Glenn’, ‘Sally’]

A List can be sorted (i.e., change its order).

- friends.sort()
  print(friends)
  [‘Glenn’, ‘Joseph’, ‘Sally’]

The sort method (unlike in strings) means “sort yourself”

- print(friends[1])
  Joseph

Built-in Functions and Lists

There are a number of functions built into Python that take Lists as parameters.

- Remember the loops we built, these are much simpler.

nums = [3, 41, 12, 9, 74, 15]

- print(len(nums))
  6
  - print(max(nums))
    74
    - print(min(nums))
      3
      - print(sum(nums))
        154
        - print(sum(nums) / len(nums))
          25.6

```
total = 0
count = 0
while True:
    inp = input('Enter a number: ')
    if inp == 'done': break
    value = float(inp)
    total = total + value
    count = count + 1

average = total / count
print('Average:', average)
```

```
numlist = list()
while True:
    inp = input('Enter a number: ')
    if inp == 'done': break
    value = float(inp)
    numlist.append(value)

average = sum(numlist) / len(numlist)
print('Average:', average)
```

,