**Lists**

Ordered sequence of elements/information accessible by index

A list is denoted by square brackets, []

Lists are python objects, everything in Python is an object, objects have data and type. Objects have methods and functions

We can access this information by object\_name.do\_something()

A list contains elements that are usually homogenous (all integers, all strings etc.) but they can contain mixed types. Lists can be changed i.e. immutable

An element of a list can be found in an index (aka position) in list, in python indices start at 0

L = [2, 1, 3]

L[1] = 5 L is still pointing to the same object in memory, it does create a new list, it simply changes the element in that index/position. We can sum up the elements of a list

total = 0

For i in range(len(L)):

total = total + L[i]

Print(total)

The range function goes from 0 to n-1

**List Operations**

.append() 🡪 Add something to the end of a list, it mutates the list

.extend() 🡪 add two lists together, it mutates the list

.del() 🡪deletes an elements from a specific position, mutates the list

.remove() 🡪 removes the specific element in the list regardless of where it is in the list, looks for the element and removes it, if it occurs multiple times, it removes the first occurrence of it, if the element is not in the list, it throws an error.

.pop() 🡪 removes the **last** element by default in the list, if you specific an index, it remove the elements at the index specified. returns 0 and mutates the list

.sort() 🡪 sorts the list but does not returns sorted list, it mutates the list

.sorted() 🡪returns sorted list, does not mutate list

.split() 🡪 convert string to list

.join() to turn to a list of characters to a string

Returns a character which is alphabetically the highest character in the string using ASCII characters