Listas

1- Declarar lista vazia:

lista\_um = [ ]

Declarar lista não vazia:

lista\_dois = [10, 20, 30, 40]

0 1 2 3 🡨 índice ou posição da lista

2- Métodos da lista:

x = 123

lista.append(*x*)

Adiciona um elemento ao final da lista. Equivalente a: a[len(a):] = [x].

lista.insert(*i*, *x*)

Insere um item (x) numa determinada posição (i).

O primeiro argumento é o índice do elemento (i) antes do qual será feita a inserção,

Ex: a.insert(0, x) insere no início da lista, e a.insert(len(a), x) é equivalente a a.append(x).

lista.remove(*x*)

Remove o primeiro item encontrado na lista cujo valor é igual a *x*.

Se não existir valor igual, uma exceção ValueError é levantada.

lista.pop([*i*])

Remove o item na posição dada e o devolve.

Se nenhum índice for especificado, a.pop() remove e devolve o último item na lista.

lista.clear()

Remove todos os itens da lista. Equivalente a del a[:].

lista.index(*x*[, *start*[, *end*]])

Retorna o índice baseado em zero na lista do primeiro item cujo valor é igual a x.

Gera um ValueError se não houver tal item.

Os argumentos opcionais início e fim são interpretados como na *slice notation* e são usados ​​para limitar a pesquisa a uma subsequência específica da lista.

O índice retornado é calculado em relação ao início da sequência completa em vez do argumento inicial.

lista.count(*x*)

Retorna o número de vezes que x aparece na lista.

lista.sort(*key=None*, *reverse=False*)

Ordena os itens da lista no lugar (os argumentos podem ser usados para uma organização personalizada, veja [sorted()](https://docs.python.org/3/library/functions.html#sorted) para a sua explicação).

lista.reverse()

Inverta os elementos da lista.

lista.copy()

Retorna uma cópia superficial da lista. Equivalente a a[:].

lista.extend(*iterable*)

Estende a lista, adicionando todos os elementos do *iterable*. Equivalente a: a[len(a):] = iterable.