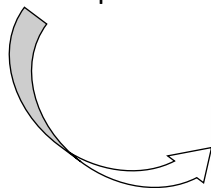


# Listas



## Lista

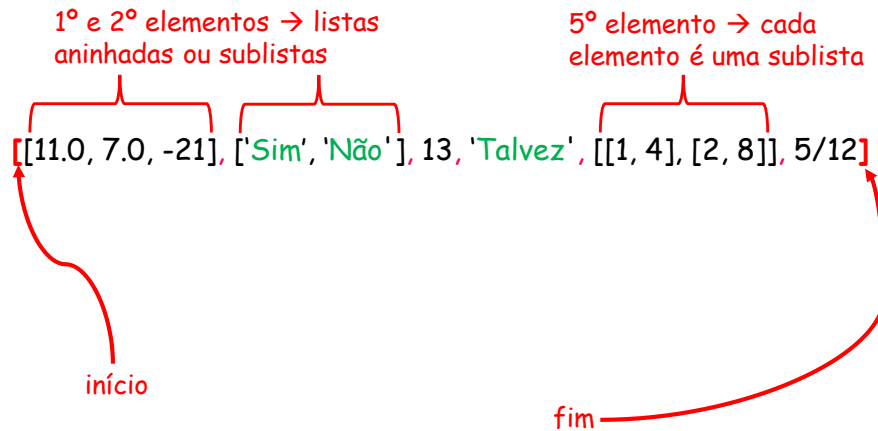
- ✓ Sequência ou coleção de valores, delimitada por colchetes
- ✓ Elementos
  - de qualquer tipo, inclusive, outras listas
  - separados por vírgulas
- ✓ Tamanho variável
  - elementos podem ser inseridos, removidos e alterados



**Listas são mutáveis!!!**



## ✓ Exemplo de lista



3

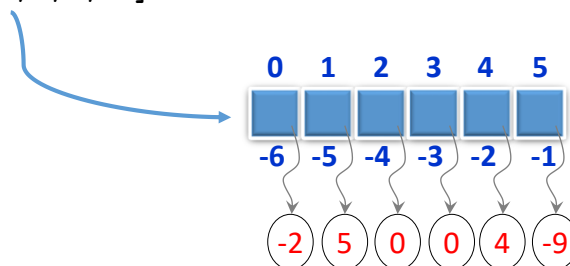


## ✓ Elementos identificados por índices (tipo inteiro)

- de 0 até (comprimento da lista) - 1
- de -1 até -(comprimento da lista)

## ✓ Exemplo

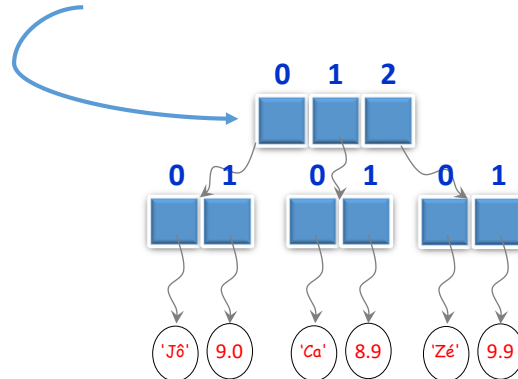
[-2, 5, 0, 0, 4, -9]



4

✓ Exemplo

[ ['Jô', 9.0], ['Ca', 8.9], ['Zé', 9.9] ]



5

✓ Criando uma lista vazia

`variávelLista = []`

ou

`variávelLista = list()`

✓ Criando por enumeração

`variávelLista = [el1, el2, ..., eln]`

6

```
amigos = ['João', 'Pedro', 'Ana']
```

```
megaSena = [1, 17, 22, 24, 25, 36]
```

```
vazia = []
```

```
mista = ['Pedro', 9.0, 15 * 2, [1, 2, 'Maya']]
```

```
listas = [amigos, megaSena]
```

7

✓ Tamanho de uma lista

**len(variávelLista)**

- retorna a quantidade de elementos da lista
- uma sublista é considerada um elemento da lista que a contém

✓ Exemplos

```
amigos = ['João', 'Pedro', 'Ana']  $\xrightarrow{\text{len(amigos)}}$  3  
megaSena = [1, 17, 22, 24, 25, 36]  $\xrightarrow{\text{len(megaSena)}}$  6  
vazia = []  $\xrightarrow{\text{len(vazia)}}$  0  
mista = ['Pedro', 9.0, 15 * 2, [1, 2, 'Maya']]  $\xrightarrow{\text{len(mista)}}$  4  
listas = [amigos, megaSena]  $\xrightarrow{\text{len(listas)}}$  2
```

8

- ✓ Exibindo o conteúdo de uma lista

`print (variávelLista)`

- ✓ Exemplos

```
print (amigos) → ['João', 'Pedro', 'Ana']
print (megaSena) → [1, 17, 22, 24, 25, 36]
print (vazia) → []
print (mista) → ['Pedro', 9.0, 30, [1, 2, 'Maya']]
print (listas) → [['João', 'Pedro', 'Ana'],
                  [1, 17, 22, 24, 25, 36]]
```

9

- ✓ Exibindo um elemento de uma lista

`print (variávelLista[índice])`

- índice inválido provoca erro

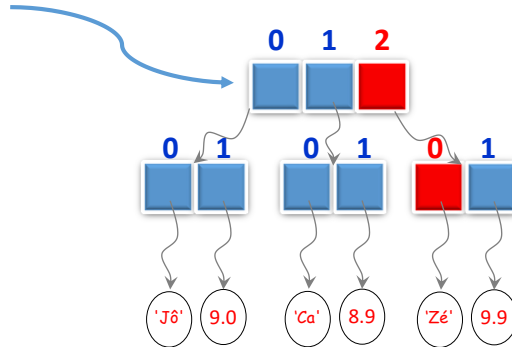
- ✓ Exemplos

```
print (amigos[2]) → Ana
print (megaSena[1]) → 17
print (mista[0]) → Pedro
print (listas[1]) → [1, 17, 22, 24, 25, 36]
print (listas[2]) → IndexError: list index out of range
```

10

- ✓ Como exibir o nome do 3º aluno da lista abaixo?

Inotas = [ ['Jô', 9.0], ['Ca', 8.9], ['Zé', 9.9] ]



`print (Inotas[2][0])` OU `print (Inotas[-1][0])`

- ✓ Alterando um elemento de uma lista

**`variávelLista[índice] = <expressão>`**

- ✓ Exemplos

`megaSena = [1, 17, 22, 24, 25, 36]`

`megaSena[4] = 31`

`megaSena[-1] = 55`

`print (megaSena)` → `[1, 17, 22, 24, 31, 55]`

`Inotas = [ ['Jô', 9.0], ['Ca', 8.9], ['Zé', 9.9] ]`

`Inotas[2][1] = 10.0`

`print (Inotas)` → `[ ['Jô', 9.0], ['Ca', 8.9], ['Zé', 10.0] ]`

- ✓ Alterando um elemento de uma lista

`variávelLista[índice] = <expressão>`

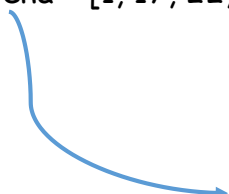
- ✓ Exemplos

```
megaSena = [1, 17, 22, 24, 31, 55]
megaSena[4] = [31, 33]
print (megaSena) → [1, 17, 22, 24, [31, 33], 55]
megaSena[0:3] = [5, 7, 11]
print (megaSena) → [5, 7, 11, 24, [31, 33], 55]
```

13

- ✓ Escrever uma função que receba uma lista já preenchida e imprima seus elementos, um em cada linha
- ✓ Exemplo

```
megaSena = [1, 17, 22, 24, 25, 36]
```

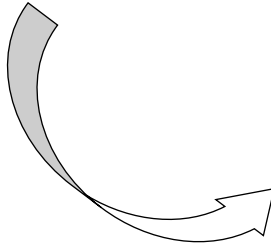


```
1
17
22
24
25
36
```

14

para cada elemento da lista

imprimir o elemento



índice = 0

enquanto índice < tamanho da lista

imprimir elemento

índice += 1

15

índice = 0

tamLista = len(lista)

enquanto índice < tamLista

imprimir lista[índice]

índice += 1

16



**# Função**

```
def imprimeLista (lista):  
    indice = 0  
    tamLista = len (lista)  
    while indice < tamLista:  
        print (lista[indice])  
        indice += 1  
    return
```

**# Teste da função**

```
megaSena = [1, 17, 22, 24, 25, 36]  
imprimeLista (megaSena)
```