
PRG203402 - Lógica de Programação: Estruturas de decisão e repetição

João Cláudio Elsen Barcellos

Engenheiro Eletricista
Formado na Universidade Federal de Santa Catarina
campus Florianópolis
joaoclaudiobarcellos@gmail.com

6 de Maio de 2025

** Créditos ao Prof. Emerson Ribeiro de Mello, o qual criou e disponibilizou o template aqui usado, via ShareLaTeX*

*** Créditos ao Prof. Renan Augusto Starke, o qual forneceu parte do conteúdo usado nesta apresentação*

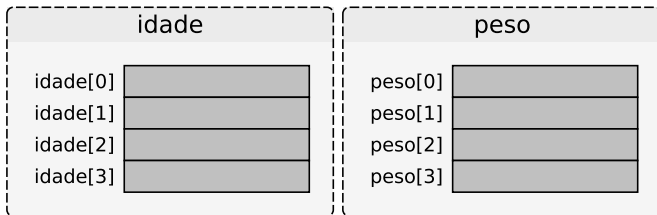


Na aula de hoje veremos...



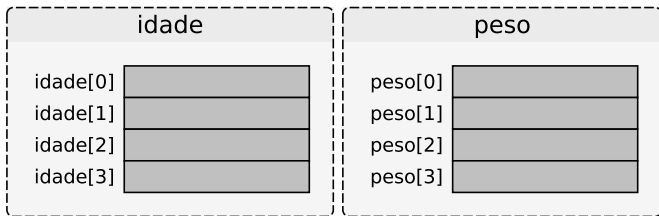
Definição

Listas podem ser definidas como uma coleção ou conjunto de dados reunidos sob um mesmo nome.



Listas

```
# Cria uma lista ``idade'' com dados inteiros  
idade = [21, 25, 36, 78]  
# Cria uma lista ``peso'' com números reais  
peso = [72.0, 74.5, 88.9, 102.0]
```



As listas em Python facilitam algumas operações, como:

- Indexação.

- Corte.

- Soma.

- Multiplicação.

- Verificação de membros.

O objetivo é agrupar dados para melhorar o acesso lidando com conjuntos dinâmicos

Listas implementam maneiras práticas para:

- encontrar o comprimento.

- encontrar o maior ou menor elemento.

- se um dado existe.

- ...



Exemplos: listas constantes

Python permite que as listas contenham dados de diferentes tipos.

```
lista_1 = ['programação', 'circuitos', 1997, 2000];  
lista_2 = [1, 2, 3, 4, 5];  
lista_3 = ["a", "b", "c", "d"];
```

```
# print exibe todos os dados das listas  
print(lista_1)  
print(lista_2)  
print(lista_3)
```

```
['programação', 'circuitos', 1997, 2000]  
[1, 2, 3, 4, 5]  
['a', 'b', 'c', 'd']
```



Exemplos: indexação

Você pode alterar qualquer elemento da lista utilizando o operador [].

```
lista_1 = ['programação', 'circuitos', 1997, 2000];  
  
# Podemos alterar itens individuais da lista (indexação)  
lista_1[0] = 'gosto de programacao'  
print(lista_1)
```

```
['gosto de programacao', 'circuitos', 1997, 2000]
```



Exemplos: processando todos os itens

Usamos o laço **for** para iterar na lista.

```
lista_1 = ['programação', 'circuitos', 1997, 2000];  
  
# Lê-se: para cada item i em lista_1  
for i in lista_1:  
    print(i)  
print(lista_1)
```

```
programação  
circuitos  
1997  
2000
```



Exemplos: acrescentar

Acrescentar um dado no final da lista.

```
a = ["abelha", "zangão"]  
print(a)  
  
a.append("formiga")  
  
print(a)
```

```
['abelha', 'zangão']  
['abelha', 'zangão', 'formiga']
```



Exemplos: estender

Estender uma lista.

```
lista_1 = ['programação', 'circuitos', 1997, 2000];  
lista_2 = [1, 2, 3, 4, 5 ];  
  
lista_1.extend(lista_2);  
  
print(lista_1)
```

```
['programação', 'circuitos', 1997, 2000, 1, 2, 3, 4, 5]
```



Exemplos: inserir

Inserir um dado.

```
a = ["abelha", "zangão"]
```

```
print(a)
```

```
# Adiciona na posição 0, início da lista
```

```
a.insert(0, "formiga")
```

```
print(a)
```

```
# Adiciona na posição 2
```

```
a.insert(2, "mosca")
```

```
print(a)
```

```
['abelha', 'zangão']
```

```
['formiga', 'abelha', 'zangão']
```

```
['formiga', 'abelha', 'mosca', 'zangão']
```



Outros métodos

Remover do fim: `pop()`

Remover todos os elementos: `clear()`

Número de vezes que um elemento aparece na lista: `count(x)`

Ordenar: `sort()`

Inverter: `reverse()`

Comprimento: `len(s)`



