



# AULA III

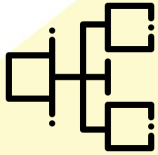
---

22/08/19

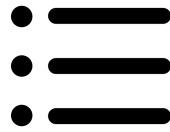
# Conteúdo

- Estrutura de Dados
- Listas
- Índices
- Acessar Valores
- Principais métodos e funções
- Listas dentro de listas
- Funções
- Parâmetros de uma função
- Escopo da função

# Sumário



Estrutura de  
Dados



Listas



Índices



Como acessar  
valores

**.append()**

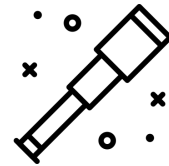
Principais  
Métodos e Funções  
com Listas



Funções



Parâmetros de  
uma função



Escopo de uma  
função



# Estrutura de Dados

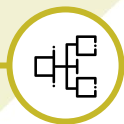
[ ] Listas

( ) Tuplas

{:} Dicionários

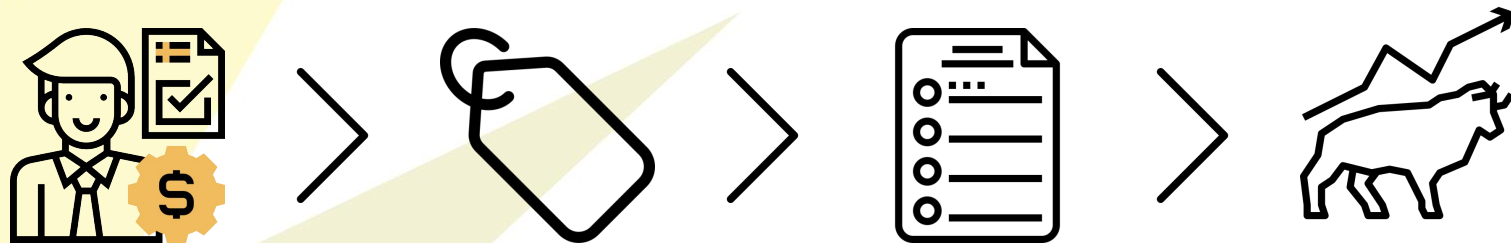
{,} Sets

*“Objeto que contém outros objetos”*



# Listas

*“Imagine que você seja um operador da bolsa de valores que esteja acompanhando a tendência de crescimento do valor de uma ação”*



*Valores dos 4 primeiros meses:*

1. R\$ 100,00      2. R\$ 110,00      3. R\$ 125,00      4. R\$ 150,50



# Listas

*Valores dos 4 primeiros meses:*

1. R\$ 100,00
2. R\$ 110,00
3. R\$ 125,00
4. R\$ 150,50

```
[1] # O aluno atribuiria o valor do 1º mês à uma variável x  
x = 100  
x
```

```
↳ 100
```

```
[2] # Agora, faria o mesmo para o 2º mês:  
x = 110  
print(x)
```

```
↳ 110
```

Entretanto, note que agora o valor 110 sumiu.

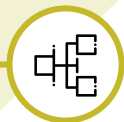
Para resolver isso o aluno poderia atribuir cada valor à uma variável diferente:

```
[3] mes_1 = 100  
    mes_2 = 110  
    mes_3 = 125  
    mes_4 = 150.50 # note que virgulas são escritas com ponto
```

Entretanto, perceba que acessar os dados se torna uma tarefa inconveniente:

```
[4] # Terei que digitar todas as variáveis  
    print(mes_1)  
    print(mes_2)  
    print(mes_3)  
    print(mes_4)
```

```
↳ 100  
    110  
    125  
    150.5
```



# Listas

*Solução:*

```
[5] cotacao = [100, 110, 125, 150.5]  
cotacao
```

```
↳ [100, 110, 125, 150.5]
```

```
[6] lista_vazia = list()  
    lista_vazia = []      # A primeira e a segunda linha fazem a mesma coisa  
    lista_vazia
```

```
↳ []
```

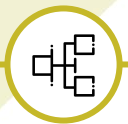
*Mas como acessar os valores?*



# Listas: Índices

**cotacao**

100	110	125	150.5
0	1	2	3





# Listas: Como acessar valores



```
[10] cotacao
```

```
↳ [100, 110, 125, 150.5]
```



```
[7] cotacao[0] [8] cotacao[1] [9] cotacao[2]
```

```
↳ 100
```

```
↳ 110
```

```
↳ 125
```



```
[13] cotacao[0:3] [14] cotacao[:2] [15] cotacao[1:] # vai d
```

```
↳ [100, 110, 125] ↳ [100, 110] ↳ [110, 125, 150.5]
```



# Listas: Atribuir valores



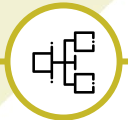
```
[17] cotacao[1] = 0  
cotacao
```

```
↳ [100, 0, 125, 150.5]
```

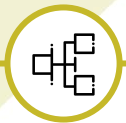


```
[18] cotacao[2:] = [0, 1]  
cotacao
```

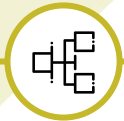
```
↳ [100, 0, 0, 1]
```



# Listas: Principais métodos e funções



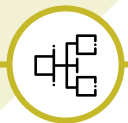
# Listas: “Nested” lists



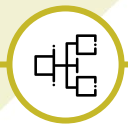
# Funções



# Funções: Parâmetros



# Funções: Escopo





# OBRIGADX!

