

Módulo 02 | Python: Estruturas de Dados

Caderno de **Aula**

Desenvolvedor [Enzo Schitini](#)

▼ Tópicos

1. Listas;
2. Conjuntos;
3. Dicionários.

▼ Aulas



▼ 1. Listas

▼ 1.1. Motivação

O aplicativo do seu banco registra toda a sua movimentação financeira. O final do dia, o app consolida o saldo final para que você possa controlar sua vida financeira.

```
dia_11_saldo_inicial = 1000
```

```
dia_11_transacao_1 = 243
dia_11_transacao_2 = -798.58
dia_11_transacao_3 = 427.12
dia_11_transacao_4 = -10.91
```

```
dia_11_saldo_final = dia_11_saldo_inicial + dia_11_transacao_1 + dia_11_transacao_2 + dia_11_transacao_3 + dia_11_transacao_4
```

```
print(dia_11_saldo_final)
```

```
860.63
```

Será que exista uma forma melhor de armazenar as transações diárias?

▼ 1.2. Definição

Armazenam sequências mutáveis e ordenadas de valores. São do tipo `list`:

```
usuario_web = ['André Perez', 'andre.perez', 'andre123', 'andre.perez@gmail.com']
```

```
print(usuario_web)
print(type(usuario_web))
```

```
['André Perez', 'andre.perez', 'andre123', 'andre.perez@gmail.com']
<class 'list'>
```

```
idade = 20
saldo_em_conta = 723.15
usuario_loggedin = True
```

```
usuario_web = ['André Perez', idade, 'andre.perez', 'andre123', 'andre.perez@gmail.com', saldo_em_conta, usuario_loggedin]
```

```
# inserir um elemento sem substituir: list.insert(index, val)
juros.insert(0, 0.10)
print(juros)
```

```
[0.1, 0.05, 0.07, 0.1, 0.04, 0.08, 0.09]
```

```
# inserir um elemento no fim da lista: list.append(val)
juros.append(0.09)
print(juros)
```

```
[0.1, 0.05, 0.07, 0.02, 0.04, 0.08, 0.09]
```

```
# remover um elemento pelo valor: list.remove(val)
juros.remove(0.1)
print(juros)
```

```
[0.05, 0.07, 0.02, 0.04, 0.08, 0.09]
```

```
# remover um elemento pelo índice: list.pop(val)
terceiro_juros = juros.pop(2)
print(terceiro_juros)
```

```
0.02
```

```
print(juros)
```

```
[0.05, 0.07, 0.04, 0.08, 0.09]
```

▼ 1.5. Conversão

Podemos converter alguns tipos de variáveis em listas, como *strings*.

```
email = 'andre.perez@gmail.com'
caracteres_email = list(email)
```

```
print(email)
print(caracteres_email)
```

```
andre.perez@gmail.com
```

```
['a', 'n', 'd', 'r', 'e', '.', 'p', 'e', 'r', 'e', 'z', '@', 'g', 'm', 'a', 'i', 'l', '.', 'c', 'o', 'm']
```

▼ 1.6. Revisitando a motivação

```
dia_11_saldo_inicial = 1000
```

```
dia_11_transacoes = []
```

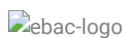
```
dia_11_transacoes.append(243)
dia_11_transacoes.append(-798.58)
dia_11_transacoes.append(427.12)
dia_11_transacoes.append(-10.91)
```

```
print(dia_11_transacoes)
```

```
[243, -798.58, 427.12, -10.91]
```

```
dia_11_saldo_final = dia_11_saldo_inicial + dia_11_transacoes[0] + dia_11_transacoes[1] + dia_11_transacoes[2] + dia_11_transacoes[3]
print(dia_11_saldo_final)
```

```
860.63
```



▼ 2. Conjuntos

▼ 2.1. Motivação

▼ 2.5. Conversão

Podemos converter conjuntos para lista e vice e versa.

```
times_paulistas = {'São Paulo', 'Palmeiras', 'Corinthians', 'Santos'}

print(times_paulistas)
print(type(times_paulistas))

{'Santos', 'São Paulo', 'Corinthians', 'Palmeiras'}
<class 'set'>

print(list(times_paulistas))
print(type(list(times_paulistas)))

['Santos', 'São Paulo', 'Corinthians', 'Palmeiras']
<class 'list'>
```

▼ 2.6. Revisitando a motivação

```
print(hashtags_semana)
print(len(hashtags_semana))

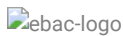
['#tiago', '#joao', '#bbb', '#sarah', '#bbb', '#fiuk', '#gil', '#thelma', '#lourdes', '#rafa', '#fora', '#danilo', '#juliete', '#ar
15
```



```
hashtags_semana = list(set(hashtags_seg + hashtags_ter + hashtags_qua + hashtags_qui + hashtags_sex))

print(hashtags_semana)
print(len(hashtags_semana))

['#fora', '#lourdes', '#gil', '#sarah', '#tiago', '#danilo', '#fiuk', '#thelma', '#rafa', '#arthur', '#bbb', '#juliete', '#joao']
13
```



▼ 3. Dicionários

▼ 3.1. Motivação

Para se conectar a uma rede wi-fi, você precisa de duas informações: o nome da rede e a senha de acesso. Quando você vai acessar uma nova rede, você encontra uma lista de redes disponíveis:

```
wifi_disponiveis = ['rede1', 'cnx_cnx', 'uai-fi', 'r3d3']
print(wifi_disponiveis)

['rede1', 'cnx_cnx', 'uai-fi', 'r3d3']
```

Você consegue identificar quais são os nome de redes e suas respectivas senhas? Talvez uma `list` não seja a melhor opção para armazenar esse tipo de dado.

▼ 3.2. Definição

Armazenam sequências no formato chave-valor. São do tipo `dict`.

```
brasil = {'capital': 'Brasília', 'idioma': 'Português', 'populacao': 210}

print(brasil)
print(type(brasil))

{'capital': 'Brasília', 'idioma': 'Português', 'populacao': 210}
<class 'dict'>
```

São métodos nativos do Python que nos ajudam a trabalhar no dia a dia com dicionários.

```
artigo = dict(
    titulo='Modulo 02 | Python: Estruturas de Dados',
    corpo='Topicos, Aulas, Listas, Conjuntos, Dicionários, ...',
    total_caracteres=1530
)

# adicionar/atualizar um elemento pelo chave-valor: dict.update(dict)
print(artigo)
artigo.update({'total_caracteres': 7850})
print(artigo)

artigo['total_caracteres'] = 7850

{'titulo': 'Modulo 02 | Python: Estruturas de Dados', 'corpo': 'Topicos, Aulas, Listas, Conjuntos, Dicionários, ...', 'total_caract
{'titulo': 'Modulo 02 | Python: Estruturas de Dados', 'corpo': 'Topicos, Aulas, Listas, Conjuntos, Dicionários, ...', 'total_caract

# remover um elemento pelo chave: dict.pop(key)
print(artigo)
total_caracteres = artigo.pop('total_caracteres')
print(artigo)

{'titulo': 'Modulo 02 | Python: Estruturas de Dados', 'corpo': 'Topicos, Aulas, Listas, Conjuntos, Dicionários, ...', 'total_caract
{'titulo': 'Modulo 02 | Python: Estruturas de Dados', 'corpo': 'Topicos, Aulas, Listas, Conjuntos, Dicionários, ...'}
```

3.5. Conversão

Podemos converter as chaves e os itens de um dicionário em uma lista.

```
artigo = dict(
    titulo='Modulo 02 | Python: Estruturas de Dados',
    corpo='Topicos, Aulas, Listas, Conjuntos, Dicionários, ...',
    total_caracteres=1530
)

chaves = list(artigo.keys())

print(chaves)
print(type(chaves))

['titulo', 'corpo', 'total_caracteres']
<class 'list'>

valores = list(artigo.values())

print(valores)
print(type(valores))

['Modulo 02 | Python: Estruturas de Dados', 'Topicos, Aulas, Listas, Conjuntos, Dicionários, ...', 1530]
<class 'list'>
```

3.6. Revisitando a motivação

```
wifi_disponiveis = []

rede = {'nome': 'rede1', 'senha': 'cnx_cnx'}
wifi_disponiveis.append(rede)

rede = {'nome': 'uai-fi', 'senha': 'r3d3'}
wifi_disponiveis.append(rede)

print(wifi_disponiveis)

[{'nome': 'rede1', 'senha': 'cnx_cnx'}, {'nome': 'uai-fi', 'senha': 'r3d3'}]
```