

# pandas

Introdução às estruturas de dados do pandas

Objetos Index

Vimos que tanto o Series quanto o DataFrame contêm um índice explícito que permite fazer referência e modificar dados. Este objeto Index é uma estrutura interessante em si, e pode ser pensado como uma matriz imutável ou como um conjunto ordenado. Essas visões têm algumas consequências interessantes nas operações disponíveis nos objetos Index.

Os objetos index do pandas armazenam os rótulos dos eixos e outros metadados (como o nome ou os nomes dos eixos). Qualquer array ou outra sequência de rótulos que você usar ao construir uma Series ou um DataFrame será internamente convertido em um Index.

# pandas

## Estruturas de dados do pandas – Objetos Index

```
In [2]: import pandas as pd
```

```
In [4]: series = pd.Series(range(5), index=['a', 'e', 'i', 'o', 'u'])
```

```
In [5]: indice = series.index
```

```
In [6]: indice
```

```
Out[6]: Index(['a', 'e', 'i', 'o', 'u'], dtype='object')
```

```
In [7]: indice[2:]
```

```
Out[7]: Index(['i', 'o', 'u'], dtype='object')
```

### Objetos Index são imutáveis.

```
In [8]: indice[0] = 'A'
```

```
-----  
TypeError                                Traceback (most recent call last)  
<ipython-input-8-bc391c5713c5> in <module>  
----> 1 indice[0] = 'A'  
  
C:\Evaldo\FerramentasDesenvolvimento\Anaconda3\lib\site-packages\pandas\core\indexes\base.py in __setitem__(self, key, value)  
    2063  
    2064     def __setitem__(self, key, value):  
-> 2065         raise TypeError("Index does not support mutable operations")  
    2066  
    2067     def __getitem__(self, key):  
  
TypeError: Index does not support mutable operations
```

# pandas

## Estruturas de dados do pandas – Objetos Index

Por serem imutáveis, é mais seguro compartilhar objetos Index entre estruturas de dados.

```
In [22]: import numpy as np
```

```
In [23]: rotulos = pd.Index(np.arange(4))
```

```
In [24]: rotulos
```

```
Out[24]: Int64Index([0, 1, 2, 3], dtype='int64')
```

```
In [25]: series = pd.Series([-30, 0, 15, 45], index=rotulos)
```

```
In [26]: series
```

```
Out[26]: 0    -30  
         1     0  
         2    15  
         3    45  
         dtype: int64
```

```
In [27]: series.index is rotulos
```

```
Out[27]: True
```

## Estruturas de dados do pandas – Objetos Index

Além de ser como os arrays, um Index também se comporta como um conjunto de tamanho fixo.

```
In [64]: dados = {'estado': ['Minas Gerais', 'Espírito Santo', 'Rio de Janeiro', 'São Paulo'], 'populacao_2000': [17891494, 3097232, 14391282, 37032403], 'populacao_2010': [19595309, 3512672, 15993583, 41252160]}
```

```
In [65]: indice = pd.Index(['Um', 'Dois', 'Três', 'Quatro'])
```

```
In [66]: print(indice.size, indice.shape, indice.ndim, indice.dtype)
```

```
4 (4,) 1 object
```

```
In [67]: frame = pd.DataFrame(dados, columns=['estado', 'populacao_2000', 'populacao_2010'], index=indice)
```

```
In [68]: frame
```

```
Out[68]:
```

	estado	populacao_2000	populacao_2010
Um	Minas Gerais	17891494	19595309
Dois	Espírito Santo	3097232	3512672
Três	Rio de Janeiro	14391282	15993583
Quatro	São Paulo	37032403	41252160

```
In [23]: 'Um' in frame.index
```

```
Out[23]: True
```

```
In [24]: 'Cinco' in frame.index
```

```
Out[24]: False
```

## Estruturas de dados do pandas – Objetos Index

Diferente dos conjuntos, um Index do pandas pode conter rótulos duplicados. A seleção com rótulos duplicados conterão todas as ocorrências destes rótulos.

```
In [36]: duplicados = pd.Index(['Um', 'Um', 'Dois', 'Três'])
In [37]: duplicados
Out[37]: Index(['Um', 'Um', 'Dois', 'Três'], dtype='object')
In [55]: frame = pd.DataFrame(dados, columns=['estado', 'populacao_2000', 'populacao_2010'], index=duplicados)
In [56]: frame
Out[56]:
```

	estado	populacao_2000	populacao_2010
Um	Minas Gerais	17891494	19595309
Um	Espírito Santo	3097232	3512672
Dois	Rio de Janeiro	14391282	15993583
Três	São Paulo	37032403	41252160

Podemos usar fatiamento:

`duplicados[1:3]`

`Index(['Um', 'Dois'], dtype='object')`



Veja algumas operações com índices.

```
In [84]: a = pd.Index([1, 2, 3, 4, 5])
```

```
In [85]: b = pd.Index([1, 5, 6, 7, 8])
```

```
In [86]: a & b # Interseção
```

```
Out[86]: Int64Index([1, 5], dtype='int64')
```

```
In [87]: a | b # União
```

```
Out[87]: Int64Index([1, 2, 3, 4, 5, 6, 7, 8], dtype='int64')
```

```
In [88]: a - b # Diferença
```

```
Out[88]: Int64Index([0, -3, -3, -3, -3], dtype='int64')
```

### Métodos e propriedades do objeto Index.

Métodos	Descrição
append	Concatena com objetos Index adicionais, gerando um novo Index.
difference(outro)	Retorna um novo índice com elementos do índice que não estão em “outro”.
intersection	Forma uma interseção de dois objetos Index
union	Forma a união de dois objetos Index
isin(valores)	Retorna um array booleano onde os valores do índice estão em “valores”
delete(loc)	Cria um novo Index com o elemento da localização passada excluído
drop(rótulos)	Cria um novo Index com a lista de rótulos passada excluída
insert(loc)	Cria um novo Index inserindo um novo item na localização informada.

# pandas

Estruturas de dados do pandas – Objetos Index

Mais informações em:

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Index.html>

# FIM