

Usando os Diferentes Contadores

```
In [391]: import pandas as pd
```

```
In [392]: s = pd.Series(list('asdeasdeasdeasdea'))  
s
```

```
Out[392]: 0      a  
          1      s  
          2      d  
          3      e  
          4      a  
          5      s  
          6      d  
          7      e  
          8      a  
          9      s  
         10      d  
         11      e  
         12      a  
         13      s  
         14      d  
         15      e  
         16      a  
dtype: object
```

Verifica cada valor que aparece pelo menos uma vez

```
In [393]: s.unique()
```

```
Out[393]: array(['a', 's', 'd', 'e'], dtype=object)
```

Conta quantas vezes cada valor aparece

```
In [394]: s.value_counts()
```

```
Out[394]: a      5  
          e      4  
          s      4  
          d      4  
dtype: int64
```

Usando os mesmos contadores para Dataframe


```
In [399]: df = pd.DataFrame({'m1': list(m1),
                             'm2': list(m2),
                             'm3': list(m3),
                             'm4': list(m4),
                             'm5': list(m5)})
```

```
In [400]: df = pd.DataFrame({'m1': dict(df.m1.value_counts()),
                             'm2': dict(df.m2.value_counts()),
                             'm3': dict(df.m3.value_counts()),
                             'm4': dict(df.m4.value_co
unts()),
                             'm5': dict(df.
m5.value_counts())})
df
```

Out[400]:

| | m1 | m2 | m3 | m4 | m5 |
|---|----|----|----|----|----|
| c | 25 | 26 | 19 | 23 | 25 |
| C | 25 | 24 | 31 | 27 | 25 |

```
In [401]: df['Faces'] = ['Cara', 'Coroa']
```

```
In [402]: df.sort_index(axis=1, inplace=True)
df
```

Out[402]:

| | Faces | m1 | m2 | m3 | m4 | m5 |
|---|-------|----|----|----|----|----|
| c | Cara | 25 | 26 | 19 | 23 | 25 |
| C | Coroa | 25 | 24 | 31 | 27 | 25 |

```
In [403]: df.sort_index(inplace=True)
df
```

Out[403]:

| | Faces | m1 | m2 | m3 | m4 | m5 |
|---|-------|----|----|----|----|----|
| C | Coroa | 25 | 24 | 31 | 27 | 25 |
| c | Cara | 25 | 26 | 19 | 23 | 25 |

Resolução proposta

```
In [404]: eventos = pd.DataFrame({'m1': list(m1),
                                   'm2': list(m2),
                                   'm3': list(m3),
                                   'm4': list(m4),
                                   'm5': list(m5)})
```

```
In [405]: moedas = pd.DataFrame(eventos)
```

```
In [406]: # para concatenar dataframes é necessário passar uma lista deles
# de mesmo tamanho e index
df = pd.DataFrame(data = ['Cara', 'Coroa'],
                  index = ['c', 'C'],
                  columns = ['Faces'])

for item in moedas:
    df = pd.concat([df, moedas[item].value_counts()],
                  axis = 1)

df
```

Out[406]:

| | Faces | m1 | m2 | m3 | m4 | m5 |
|---|-------|----|----|----|----|----|
| c | Cara | 25 | 26 | 19 | 23 | 25 |
| C | Coroa | 25 | 24 | 31 | 27 | 25 |

Passo a passo

```
In [407]: df = pd.DataFrame(data = ['Cara', 'Coroa'],
                          index = ['c', 'C'],
                          columns = ['Faces'])

df
```

Out[407]:

| | Faces |
|---|-------|
| c | Cara |
| C | Coroa |

Concatenando lado a lado

```
In [408]: for i in moedas:
df = pd.concat([df, moedas[i].value_counts()], axis=1)

df
```

Out[408]:

| | Faces | m1 | m2 | m3 | m4 | m5 |
|---|-------|----|----|----|----|----|
| c | Cara | 25 | 26 | 19 | 23 | 25 |
| C | Coroa | 25 | 24 | 31 | 27 | 25 |

Utilizando outro index como exemplo

```
In [409]: n1 = 'AAAAAAAAAAAAAAAAAAAAAAAAAAAAAa'
n2 = 'AAAAAAAAAAAAAAAAAAAAAAAAAAAAAaAa'
n3 = 'aaaaaaaaaaaaAAAAaaaaAAAAaaaaaa'
```

```
In [410]: events = pd.DataFrame({'n1': list(n1),
                                'n2': list(n2),
                                'n3': list(n3)})
```

```
In [411]: df2 = pd.DataFrame(data = ['Maiúsc', 'Minúsc'], index = ['A', 'a'], columns =
    ['Letra'])
df2
```

```
Out[411]:
```

| | Letra |
|---|--------|
| A | Maiúsc |
| a | Minúsc |

```
In [412]: for lista in events:
    df2 = pd.concat([df2, events[lista].value_counts()], axis=1)
df2
```

```
Out[412]:
```

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| A | Maiúsc | 14 | 14 | 8 |
| a | Minúsc | 17 | 17 | 23 |

- não importa a ordem que colocarmos o index
- o pandas relacionará automaticamente o index com o dicionário passado
- repare o exemplo abaixo:

```
In [413]: events.n1.value_counts()
```

```
Out[413]: a    17
A    14
Name: n1, dtype: int64
```

Repare agora como ficaria o dataframe se invertêssemos a ordem do index

```
In [414]: df3 = pd.DataFrame(data = ['Minúsc', 'Maiúsc'], index = ['a', 'A'], columns =
    ['Letra'])
df3
```

```
Out[414]:
```

| | Letra |
|---|--------|
| a | Minúsc |
| A | Maiúsc |

```
In [415]: for item in events:
          df3 = pd.concat([df3, events[item].value_counts()], axis=1)
          df3
```

Out[415]:

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| a | Minúsc | 17 | 17 | 23 |
| A | Maiúsc | 14 | 14 | 8 |

```
In [416]: df2
```

Out[416]:

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| A | Maiúsc | 14 | 14 | 8 |
| a | Minúsc | 17 | 17 | 23 |

Podemos também reorganizar o index a qualquer momento

```
In [420]: # Lembrando que para fixar o método, deve-se utilizar o inplace=True
          # no python a letra maiúscula sempre vem primeiro na ordem alfabética
          df3.sort_index()
```

Out[420]:

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| A | Maiúsc | 14 | 14 | 8 |
| a | Minúsc | 17 | 17 | 23 |

```
In [419]: df3
```

Out[419]:

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| a | Minúsc | 17 | 17 | 23 |
| A | Maiúsc | 14 | 14 | 8 |

```
In [423]: df2
```

Out[423]:

| | Letra | n1 | n2 | n3 |
|---|--------|----|----|----|
| A | Maiúsc | 14 | 14 | 8 |
| a | Minúsc | 17 | 17 | 23 |

```
In [424]: # df2 já esta na ordem crescente, portanto o método não tem efeito  
df2.sort_index()
```

Out[424]:

| | Letra | n1 | n2 | n3 |
|----------|--------|----|----|----|
| A | Maiúsc | 14 | 14 | 8 |
| a | Minúsc | 17 | 17 | 23 |

```
In [422]: # para trocar a ordem de crescente para decrescente usa-se ascending=False  
df2.sort_index(ascending=False)
```

Out[422]:

| | Letra | n1 | n2 | n3 |
|----------|--------|----|----|----|
| a | Minúsc | 17 | 17 | 23 |
| A | Maiúsc | 14 | 14 | 8 |