

Criando Faixas de Valor

```
In [1]: import pandas as pd
```

```
In [11]: dados = pd.read_csv('../dados/aluguel12.csv', sep=';')
dados.head(10)
```

Out[11]:

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU	Valor m2
0	Apartamento	Centro	1	0	0	15	800.0	390.0	20.0	53.33
1	Apartamento	Higienópolis	1	0	0	48	800.0	230.0	0.0	16.67
2	Apartamento	Cachambi	2	0	0	50	1300.0	301.0	17.0	26.00
3	Apartamento	Grajaú	2	1	0	70	1500.0	642.0	74.0	21.43
4	Apartamento	Lins de Vasconcelos	3	1	1	90	1500.0	455.0	14.0	16.67
5	Apartamento	Copacabana	1	0	1	40	2000.0	561.0	50.0	50.00
6	Apartamento	Freguesia (Jacarepaguá)	3	0	0	54	950.0	300.0	28.0	17.59
7	Apartamento	Barra da Tijuca	2	1	1	67	1700.0	589.0	147.0	25.37
8	Apartamento	Tijuca	2	1	0	110	1900.0	700.0	138.0	17.27
9	Apartamento	Olaria	3	1	0	68	1000.0	670.0	0.0	14.71

Distribuição de Frequência

Faixas de Frequência de Quartos

- 1 e 2
- 3 e 4
- 5 e 6
- 7 ou mais

```
In [28]: # verificando o número máximo de quartos que aparece
dados['Quartos'].max()
```

Out[28]: 14

Criando Classes

```
In [36]: # criamos as classes com o valor mínimo, máximo e os limites
classes = [0, 2, 4, 6, 14]
```

Categorizando as classes

```
In [37]: # utiliza-se 2 argumentos
# nesse caso está sendo utilizado uma series e uma lista com os limites
quartos = pd.cut(dados.Quartos, classes)
quartos
```

```
Out[37]: 0      (0.0, 2.0]
1      (0.0, 2.0]
2      (0.0, 2.0]
3      (0.0, 2.0]
4      (2.0, 4.0]
...
19826  (0.0, 2.0]
19827  (0.0, 2.0]
19828  (0.0, 2.0]
19829  (0.0, 2.0]
19830      NaN
Name: Quartos, Length: 19831, dtype: category
Categories (4, interval[int64]): [(0, 2] < (2, 4] < (4, 6] < (6, 14]]
```

```
In [40]: # o parênteses não inclui o valor
# o colchetes inclui o valor
# ficando as faixas de quartos conforme estabelecido acima
pd.value_counts(quartos)
```

```
Out[40]: (0, 2]      11074
(2, 4]      8094
(4, 6]       470
(6, 14]       36
Name: Quartos, dtype: int64
```

Modificando a visualização

```
In [41]: labels = ['1 e 2 quartos', '3 e 4 quartos', '5 e 6 quartos', '7 quartos ou mais']
```

```
In [45]: # pode-se adicionar mais um parâmetro
# nesse caso utilizamos o labels
quartos = pd.cut(dados.Quartos, classes, labels=labels)
```

```
In [44]: pd.value_counts(quartos)
```

```
Out[44]: 1 e 2 quartos      11074  
         3 e 4 quartos      8094  
         5 e 6 quartos       470  
         7 quartos ou mais    36  
         Name: Quartos, dtype: int64
```

- sabendo que é possível que não tenha sido informado o número de quartos, essa informação pode ser 0
- podemos incluir então um quarto parâmetro

```
In [46]: # include_lowest inclui o menor valor possível que aparece  
quartos = pd.cut(dados.Quartos, classes, labels=labels, include_lowest=True)
```

```
In [48]: # pode-se notar que há uma diferença na faixa 1 e 2 quartos  
# ou seja, foi incluído o valor 0 também  
pd.value_counts(quartos)
```

```
Out[48]: 1 e 2 quartos      11231  
         3 e 4 quartos      8094  
         5 e 6 quartos       470  
         7 quartos ou mais    36  
         Name: Quartos, dtype: int64
```