Biblioteca Pandas

- Definição;
- Series;
- Data Frame;
- Leitura de Dados;
- Limpeza de dados sujos;
- Seleção de dados;
- Manipulação de dados;
- Visualização de dados.

Definição

- Pandas é uma biblioteca que fornece estrutura de dados adicionais para trabalhar com conjuntos de dados em Python(Grus, 2016);

- Biblioteca apropriada para analisar dividir, agrupar, manipular e limpar conjunto de dados.

Series

Series

- Array rotulado unidimensional;

- Inteiros, strings, números de ponto flutuante, objetos, etc;

- Rótulos dos eixos são referidos coletivamente como índice.

Series

```
In [13]: s = pd.Series([2,4,6,8,10])
```

```
In [14]: s

Out[14]: 0     2
         1     4
         2     6
         3     8
         4    10
         dtype: int64
```

```
In [16]: s = pd.Series([2,4,6,8,10],['a','b','c','d','e'])
```

```
In [17]: s

Out[17]: a     2
         b     4
         c     6
         d     8
         e    10
         dtype: int64
```

Data Frame

- Estrutura de dados em duas dimensões;
- Colunas com diferentes tipos de colunas;
- Operações aritméticas em linhas e colunas.

Data Frame

Data Frame

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

```
In [4]: data = {'Name':['Tom', 'Jack', 'Steve', 'Ricky'],'Age':[28,34,29,42]}
        df = pd.DataFrame(data, index=['rank1','rank2','rank3','rank4'])
        df
```

Out[4]:

	Name	Age
rank1	Tom	28
rank2	Jack	34
rank3	Steve	29
rank4	Ricky	42

Leitura de Dados

Leitura em CSV

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

```
In [8]: df = pd.read_csv("dados.csv")
```

```
In [9]: df
```

Out[9]:

	condominio	quartos	suites	vagas	area	bairro	preco	pm2
0	350	1	0.0	1.0	21	Botafogo	340000	16190.48
1	800	1	0.0	1.0	64	Botafogo	770000	12031.25
2	674	1	0.0	1.0	61	Botafogo	600000	9836.07
3	700	1	1.0	1.0	70	Botafogo	700000	10000.00
4	440	1	0.0	1.0	44	Botafogo	515000	11704.55
5	917	1	1.0	1.0	60	Botafogo	630000	10500.00
6	850	1	1.0	1.0	65	Botafogo	740000	11384.62
7	350	1	1.0	1.0	43	Botafogo	570000	13255.81

Seleção de Dados

Retorno de dtypes

```
In [10]: ## Verificar as variáveis
         df.dtypes

Out[10]: condominio      int64
         quartos         int64
         suites        float64
         vagas         float64
         area            int64
         bairro         object
         preco           int64
         pm2           float64
         dtype: object
```

```
In [13]: ## Selecionar as primeiras 5 linhas da coluna bairro
         df['bairro'][:5]

Out[13]: 0    Botafogo
         1    Botafogo
         2    Botafogo
         3    Botafogo
         4    Botafogo
         Name: bairro, dtype: object
```

Seleção de dados

In [17]:
```
##Selecionar as 5 primeiras linhas no bairro botafogo com 3 quartos
df_bairro = df['bairro'] == "Botafogo"
df_3 = df['quartos'] == 3
df[df_bairro & df_3][:5]
```

Out[17]:

	condominio	quartos	suites	vagas	area	bairro	preco	pm2
63	1942	3	1.0	2.0	129	Botafogo	1659000	12860.47
64	1200	3	1.0	1.0	120	Botafogo	1100000	9166.67
65	890	3	1.0	2.0	109	Botafogo	950000	8715.60
66	1400	3	1.0	1.0	94	Botafogo	950000	10106.38
67	1580	3	1.0	1.0	215	Botafogo	1890000	8790.70

In [22]:
```
## Ordenar as colunas
df[df_bairro & df_3][['bairro','preco','quartos','vagas']][:5]
```

Out[22]:

	bairro	preco	quartos	vagas
63	Botafogo	1659000	3	2.0
64	Botafogo	1100000	3	1.0
65	Botafogo	950000	3	2.0
66	Botafogo	950000	3	1.0
67	Botafogo	1890000	3	1.0

Limpeza de Dados

Função Replace

```
In [34]: df2 = df.head()
         df2 = df2.replace({"pm2":{12031.25: np.nan}})
         df2
```

Out[34]:

	condominio	quartos	suites	vagas	area	bairro	preco	pm2
0	350	1	0.0	1.0	21	Botafogo	340000	16190.48
1	800	1	0.0	1.0	64	Botafogo	770000	NaN
2	674	1	0.0	1.0	61	Botafogo	600000	9836.07
3	700	1	1.0	1.0	70	Botafogo	700000	10000.00
4	440	1	0.0	1.0	44	Botafogo	515000	11704.55

Função Dropna

```
In [35]: df2.dropna()
```

Out[35]:

	condominio	quartos	suites	vagas	area	bairro	preco	pm2
0	350	1	0.0	1.0	21	Botafogo	340000	16190.48
2	674	1	0.0	1.0	61	Botafogo	600000	9836.07
3	700	1	1.0	1.0	70	Botafogo	700000	10000.00
4	440	1	0.0	1.0	44	Botafogo	515000	11704.55

Manipulação de dados

Função count

```
In [23]: df["bairro"].value_counts()

Out[23]: Copacabana    346
         Tijuca        341
         Botafogo      307
         Ipanema       281
         Leblon        280
         Grajaú        237
         Gávea         205
         Name: bairro, dtype: int64
```

Função mean

```
In [28]: df.groupby("bairro").mean()
```

Out[28]:

bairro	condominio	quartos	suites	vagas	area	preco	pm2
Botafogo	914.475570	2.107492	1.048860	1.159609	83.837134	1.010614e+06	12034.486189
Copacabana	991.861272	2.101156	1.034682	1.080925	101.855491	1.216344e+06	11965.298699
Grajaú	619.940928	2.097046	0.970464	1.130802	79.949367	4.788869e+05	6145.624473
Gávea	985.234146	2.058537	1.029268	1.200000	88.497561	1.454571e+06	16511.582780
Ipanema	1357.120996	2.181495	1.192171	1.220641	100.615658	2.033096e+06	19738.407794
Leblon	1260.010714	2.207143	1.064286	1.164286	91.832143	1.946193e+06	20761.351036
Tijuca	681.175953	2.131965	0.944282	1.143695	81.457478	5.750780e+05	7149.804985

Manipulação de dados

Função Groupby

```
In [30]: df.groupby("bairro").mean()["pm2"].sort_values()

Out[30]: bairro
         Grajaú          6145.624473
         Tijuca          7149.804985
         Copacabana     11965.298699
         Botafogo       12034.486189
         Gávea          16511.582780
         Ipanema        19738.407794
         Leblon         20761.351036
         Name: pm2, dtype: float64
```

Visualização de Dados

Função Matplotlib

```
In [38]: import matplotlib.pyplot as plt
         df["bairro"].value_counts().plot.bar()
```

```
Out[38]: <matplotlib.axes._subplots.AxesSubplot at 0x9133b70>
```