

# pandas - "A melhor biblioteca do Python para Dados"

- Quase sempre que você for trabalhar com dados no Python você vai usar o pandas, então é importante você saber trabalhar com essa biblioteca muito bem
- Vamos aprender os principais comandos e os princípios básicos do pandas
- Faça o download do Gabarito desse arquivo e use como uma cartilha de consulta (Link na descrição)

## Importando o pandas

```
In [33]: ▶ import pandas as pd
```

## Criando um dataframe a partir de um dicionário

```
In [34]: ▶ # dataframe = pd.DataFrame()
venda = {'data': ['15/02/2021', '16/02/2021'],
        'valor': [500, 300],
        'produto': ['feijao', 'arroz'],
        'qtde': [50, 70],
        }
vendas_df = pd.DataFrame(venda)
```

## Visualização dos Dados

- print
- display

```
In [35]: ▶ display(vendas_df)
```

	data	valor	produto	qtde
0	15/02/2021	500	feijao	50
1	16/02/2021	300	arroz	70

## Importando arquivos e bases de dados

```
In [36]: vendas_df = pd.read_excel("Vendas.xlsx")
display(vendas_df)
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final
0	1	2019-01-01	Iguatemi Esplanada	Sapato Estampa	1	358	358
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71
...	...	...	...	...	...	...	...
93905	65012	2019-11-30	Shopping Vila Velha	Mochila Xadrez	2	283	566
93906	65013	2019-11-30	Ribeirão Shopping	Pulseira Listrado	2	79	158
93907	65013	2019-11-30	Ribeirão Shopping	Cueca Listrado	3	67	201
93908	65014	2019-11-30	Shopping Morumbi	Pulseira Linho	5	114	570
93909	65014	2019-11-30	Shopping Morumbi	Casaco Xadrez	4	259	1036

93910 rows × 7 columns

## Resumos de Visualização de Dados simples e úteis

- head
- shape
- describe

```
In [37]: ▶ display(vendas_df.head(10))
print(vendas_df.shape)
display(vendas_df.describe())
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final
0	1	2019-01-01	Iguatemi Esplanada	Sapato Estampa	1	358	358
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71
5	3	2019-01-02	Rio Mar Shopping Fortaleza	Cinto Linho	1	248	248
6	5	2019-01-02	Shopping Barra	Calça	1	170	170
7	6	2019-01-02	Shopping Ibirapuera	Polo Listrado	4	149	596
8	7	2019-01-02	Norte Shopping	Camisa Gola V Listrado	1	116	116
9	7	2019-01-02	Norte Shopping	Camisa Liso	1	105	105

(93910, 7)

	Código Venda	Quantidade	Valor Unitário	Valor Final
count	93910.000000	93910.000000	93910.000000	93910.000000
mean	32464.762155	2.166553	191.725886	414.862656
std	18809.007093	1.258732	145.215519	434.846228
min	1.000000	1.000000	30.000000	30.000000
25%	16204.000000	1.000000	100.000000	156.000000
50%	32367.000000	2.000000	155.000000	274.000000
75%	48793.750000	3.000000	248.000000	524.000000
max	65014.000000	5.000000	750.000000	3750.000000

## Pegar 1 coluna (e os pd.Series)

```
In [38]: ▶ produtos = vendas_df[['Produto', 'ID Loja']]  
display(produtos)
```

	Produto	ID Loja
0	Sapato Estampa	Iguatemi Esplanada
1	Camiseta	Iguatemi Esplanada
2	Sapato Xadrez	Iguatemi Esplanada
3	Relógio	Norte Shopping
4	Chinelo Liso	Norte Shopping
...	...	...
93905	Mochila Xadrez	Shopping Vila Velha
93906	Pulseira Listrado	Ribeirão Shopping
93907	Cueca Listrado	Ribeirão Shopping
93908	Pulseira Linho	Shopping Morumbi
93909	Casaco Xadrez	Shopping Morumbi

93910 rows × 2 columns

## .loc, um método muito importante

- Pegar 1 linha
- Pegar linhas de acordo com alguma condição
- Pegar linhas e colunas específicas
- Pegar 1 valor específico

```
In [39]: # pegar uma linha
display(vendas_df.loc[1:5])

# pegar linhas que correspondem a uma condição
vendas_norteshopping_df = vendas_df.loc[vendas_df['ID Loja'] == 'Norte Shopping']

# pegar várias linhas e colunas usando o loc
vendas_norteshopping_df = vendas_df.loc[vendas_df['ID Loja'] == 'Norte Shopping']
display(vendas_norteshopping_df)

# pegar 1 valor específico
print(vendas_df.loc[1, 'Produto'])
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71
5	3	2019-01-02	Rio Mar Shopping Fortaleza	Cinto Linho	1	248	248

	ID Loja	Produto	Quantidade
3	Norte Shopping	Relógio	3
4	Norte Shopping	Chinelo Liso	1
8	Norte Shopping	Camisa Gola V Listrado	1
9	Norte Shopping	Camisa Liso	1
100	Norte Shopping	Cueca Xadrez	5
...	...	...	...
93716	Norte Shopping	Polo Xadrez	1
93754	Norte Shopping	Calça Liso	1
93755	Norte Shopping	Casaco Estampa	2
93756	Norte Shopping	Gorro	2
93852	Norte Shopping	Bermuda Linho	1

3924 rows × 3 columns

Camiseta

## Adicionar 1 coluna

```
In [40]: ▶ # a partir de uma coluna que existe
vendas_df['Comissão'] = vendas_df['Valor Final'] * 0.05
# display(vendas_df)

# criar uma coluna com valor padrão
vendas_df.loc[:, "Imposto"] = 0
display(vendas_df)
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final	Comissão	Imposto
0	1	2019-01-01	Iguatemi Esplanada	Sapato Estampa	1	358	358	17.90	0
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360	18.00	0
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368	18.40	0
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600	30.00	0
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71	3.55	0
...	...	...	...	...	...	...	...	...	...
93905	65012	2019-11-30	Shopping Vila Velha	Mochila Xadrez	2	283	566	28.30	0
93906	65013	2019-11-30	Ribeirão Shopping	Pulseira Listrado	2	79	158	7.90	0
93907	65013	2019-11-30	Ribeirão Shopping	Cueca Listrado	3	67	201	10.05	0
93908	65014	2019-11-30	Shopping Morumbi	Pulseira Linho	5	114	570	28.50	0
93909	65014	2019-11-30	Shopping Morumbi	Casaco Xadrez	4	259	1036	51.80	0

93910 rows × 9 columns

## Adicionar 1 linha

- Linhas de um complemento da base de dados

```
In [41]: > vendas_dez_df = pd.read_excel("Vendas - Dez.xlsx")
#display(vendas_dez_df)

vendas_df = vendas_df.append(vendas_dez_df)
display(vendas_df)
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final	Comissão	Imposto
0	1	2019-01-01	Iguatemi Esplanada	Sapato Estampa	1	358	358	17.90	0.0
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360	18.00	0.0
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368	18.40	0.0
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600	30.00	0.0
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71	3.55	0.0
...	...	...	...	...	...	...	...	...	...
7084	69996	2019-12-26	Center Shopping Uberlândia	Short Listrado	2	102	204	NaN	NaN
7085	69996	2019-12-26	Center Shopping Uberlândia	Mochila	4	270	1080	NaN	NaN
7086	69996	2019-12-26	Center Shopping Uberlândia	Pulseira Estampa	1	87	87	NaN	NaN
7087	69997	2019-12-26	Ribeirão Shopping	Camisa Listrado	1	108	108	NaN	NaN
7088	69997	2019-12-26	Ribeirão Shopping	Short Linho	2	133	266	NaN	NaN

100999 rows × 9 columns

## Excluir linhas e colunas

```
In [43]: vendas_df = vendas_df.drop(0, axis=0)
display(vendas_df)
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final	Comissão
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360	18.00
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368	18.40
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600	30.00
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71	3.55
5	3	2019-01-02	Rio Mar Shopping Fortaleza	Cinto Linho	1	248	248	12.40
...	...	...	...	...	...	...	...	...
7084	69996	2019-12-26	Center Shopping Uberlândia	Short Listrado	2	102	204	NaN
7085	69996	2019-12-26	Center Shopping Uberlândia	Mochila	4	270	1080	NaN
7086	69996	2019-12-26	Center Shopping Uberlândia	Pulseira Estampa	1	87	87	NaN
7087	69997	2019-12-26	Ribeirão Shopping	Camisa Listrado	1	108	108	NaN
7088	69997	2019-12-26	Ribeirão Shopping	Short Linho	2	133	266	NaN

100997 rows × 8 columns

## Valores Vazios

- Deletar linhas/colunas vazias
- Deletar linhas que possuem valores vazios
- Preencher valores vazios (média e último valor)



```
In [44]: # deletar linhas e colunas completamente vazias
# vendas_df = vendas_df.dropna(how='all', axis=1)

# deletar linhas que possuem pelo menos 1 valor vazio
# vendas_df = vendas_df.dropna()

# preencher valores vazios
# preencher com a média da coluna
vendas_df['Comissão'] = vendas_df['Comissão'].fillna(vendas_df['Comissão'].mean())
display(vendas_df)

# preencher com o último valor
vendas_df = vendas_df.ffill()
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final	Comissão
1	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360	18.000000
2	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368	18.400000
3	2	2019-01-02	Norte Shopping	Relógio	3	200	600	30.000000
4	2	2019-01-02	Norte Shopping	Chinelo Liso	1	71	71	3.550000
5	3	2019-01-02	Rio Mar Shopping Fortaleza	Cinto Linho	1	248	248	12.400000
...	...	...	...	...	...	...	...	...
7084	69996	2019-12-26	Center Shopping Uberlândia	Short Listrado	2	102	204	20.743163
7085	69996	2019-12-26	Center Shopping Uberlândia	Mochila	4	270	1080	20.743163
7086	69996	2019-12-26	Center Shopping Uberlândia	Pulseira Estampa	1	87	87	20.743163
7087	69997	2019-12-26	Ribeirão Shopping	Camisa Listrado	1	108	108	20.743163
7088	69997	2019-12-26	Ribeirão Shopping	Short Linho	2	133	266	20.743163

100997 rows × 8 columns

## Calcular Indicadores

- Groupby
- Value Counts



```
In [49]: # value counts
transacoes_loja = vendas_df['ID Loja'].value_counts()
display(transacoes_loja)

# group by
faturamento_produto = vendas_df[['Produto', 'Valor Final']].groupby('Produto')
display(faturamento_produto)
```

```
Shopping Vila Velha          4234
Palladium Shopping Curitiba  4210
Norte Shopping               4179
Rio Mar Shopping Fortaleza   4118
Bourbon Shopping SP          4116
Iguatemi Campinas           4108
Rio Mar Recife               4099
Shopping Center Leste Aricanduva 4093
Shopping SP Market           4080
Shopping Ibirapuera          4051
Novo Shopping Ribeirão Preto  4049
Ribeirão Shopping            4048
Salvador Shopping            4030
Shopping Iguatemi Fortaleza   4021
Shopping Center Interlagos    4021
Center Shopping Uberlândia    4013
Shopping Eldorado             4002
Passei das Águas Shopping     4000
Shopping União de Osasco      3995
Iguatemi Esplanada           3979
Shopping Barra                3962
Shopping Morumbi              3959
Parque Dom Pedro Shopping     3902
Shopping Recife               3891
Shopping Midway Mall          3837
Name: ID Loja, dtype: int64
```

	Valor Final
Produto	
Bermuda	272250
Bermuda Estampa	291694
Bermuda Linho	394680
Bermuda Liso	275692
Bermuda Listrado	293237
...	...
Tênis Estampa	457728
Tênis Linho	538608
Tênis Liso	474544
Tênis Listrado	481032
Tênis Xadrez	459725

120 rows × 1 columns

## Mesclar 2 dataframes (Procurar informações de um dataframe em outro)

```
In [51]: gerentes_df = pd.read_excel("Gerentes.xlsx")
# display(gerentes_df)

vendas_df = vendas_df.merge(gerentes_df)
display(vendas_df)
```

	Código Venda	Data	ID Loja	Produto	Quantidade	Valor Unitário	Valor Final	Comissão	Gerente
0	1	2019-01-01	Iguatemi Esplanada	Camiseta	2	180	360	18.000000	Salvador
1	1	2019-01-01	Iguatemi Esplanada	Sapato Xadrez	1	368	368	18.400000	Salvador
2	21	2019-01-02	Iguatemi Esplanada	Camisa Gola V Listrado	2	116	232	11.600000	Salvador
3	34	2019-01-02	Iguatemi Esplanada	Sapato Listrado	1	363	363	18.150000	Salvador
4	34	2019-01-02	Iguatemi Esplanada	Camisa Gola V Liso	2	118	236	11.800000	Salvador
...	...	...	...	...	...	...	...	...	...
100992	69846	2019-12-25	Salvador Shopping	Short Estampa	2	96	192	20.743163	Mateus
100993	69846	2019-12-25	Salvador Shopping	Tênis Estampa	5	256	1280	20.743163	Mateus
100994	69850	2019-12-25	Salvador Shopping	Calça Estampa	4	177	708	20.743163	Mateus
100995	69972	2019-12-26	Salvador Shopping	Terno Liso	3	720	2160	20.743163	Mateus
100996	69972	2019-12-26	Salvador Shopping	Sapato Estampa	5	358	1790	20.743163	Mateus

100997 rows × 9 columns

## Como continuar aprendendo?

## Minicurso Gratuito de Análise de Dados com Python na Descrição

