

Tutorial 03 - Different Ways Of Creating DataFrame # Primeiramente vamos importar o arquivo com extensão .xlsx escolhendo a aba # Google: 'pandas IO'

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

```
In [2]: df_data = pd.read_excel('sample_data_tutorial_03.xlsx','data')
df_data
```

Out[2]:

	day	event	windspeed[kts]	temp.[F]
0	2017-01-01	Rain	6	32
1	2017-01-02	Sunny	7	35
2	2017-01-03	Snow	2	28
3	2017-01-04	Snow	7	24
4	2017-01-05	Rain	4	32
5	2017-01-06	Sunny	2	32

```
In [3]: df_data1 = pd.read_excel('sample_data_tutorial_03.xlsx','data1')
df_data1
```

Out[3]:

	day	windspeed[kts]	event	temp.[F]
0	2017-01-01	12	Rain	39
1	2017-01-02	14	Sunny	42
2	2017-01-03	4	Snow	35
3	2017-01-04	14	Snow	31
4	2017-01-05	8	Rain	39
5	2017-01-06	4	Sunny	39

Tutorial 04 - Read Write Excel CSV File

```
In [4]: # Neste comando de leitura iremos pular a primeira linha do arquivo .csv usando o comando "skiprows"
# Um comando similar seria utilizar "header=1". Isto significa que o título está na linha 2, lembrar que o index começa em zero!
# Para maiores detalhes das opções, google: "pandas read_csv"
df4 = pd.read_csv('sample_data_tutorial_04.csv', skiprows=1)
df4
```

Out[4]:

	tickers	eps	revenue	price	people
0	GOOGL	27.82	87	845	larry page
1	WALMART	4.61	484	65	n.a.
2	MSFT	-1	85	64	bill gates
3	RIL	not available	50	1023	mukeshi ambani
4	TATA	5.6	-1	n.a.	ratan tata

```
In [5]: # Imagine agora que o arquivo iniciasse com o dados e não tivesse "header"
df4a = pd.read_csv('sample_data_tutorial_04a.csv', header=None)
df4a
```

Out[5]:

	0	1	2	3	4
0	GOOGL	27.82	87	845	larry page
1	WALMART	4.61	484	65	n.a.
2	MSFT	-1	85	64	bill gates
3	RIL	not available	50	1023	mukeshi ambani
4	TATA	5.6	-1	n.a.	ratan tata

```
In [6]: # Imagine o caso anterior mas com a inserção de nomes
df4b = pd.read_csv('sample_data_tutorial_04a.csv', header=None, names=["ticker","eps","revenue","price","people"])
df4b
```

Out[6]:

	ticker	eps	revenue	price	people
0	GOOGL	27.82	87	845	larry page
1	WALMART	4.61	484	65	n.a.
2	MSFT	-1	85	64	bill gates
3	RIL	not available	50	1023	mukeshi ambani
4	TATA	5.6	-1	n.a.	ratan tata

```
In [7]: # Imagine que o csv é muito grande e queremos somente as 3 primeiras linhas
df4c = pd.read_csv('sample_data_tutorial_04.csv', skiprows=1, nrows=3)
df4c
```

Out[7]:

	tickers	eps	revenue	price	people
0	GOOGL	27.82	87	845	larry page
1	WALMART	4.61	484	65	n.a.
2	MSFT	-1.00	85	64	bill gates

```
In [8]: # Retornando ao arquivo inicial, para substituir algumas células para "NaN" (Not a
        Number)
df4 = pd.read_csv('sample_data_tutorial_04.csv', skiprows=1, na_values=["not availa
ble", "n.a."])
df4
```

Out[8]:

	tickers	eps	revenue	price	people
0	GOOGL	27.82	87	845.0	larry page
1	WALMART	4.61	484	65.0	NaN
2	MSFT	-1.00	85	64.0	bill gates
3	RIL	NaN	50	1023.0	mukeshi ambani
4	TATA	5.60	-1	NaN	ratan tata

```
In [9]: # Verificamos nos dados anteriores que o revenue da TATA está negativo e tem que se
        r zero ou maior que zero.
        # Se colocassemos no filtro anterior o "-1" ele também alteraria o valor "-1" do ep
        s.
        # Para utilizar diferentes filtros nas colunas utilizar um dicionário:
df4 = pd.read_csv('sample_data_tutorial_04.csv', skiprows=1, na_values={
    'eps': ["not available", "n.a."],
    'revenue': ["not available", "n.a.", -1],
    'price': ["not available", "n.a."],
    'people': ["not available", "n.a."]
})
df4
```

Out[9]:

	tickers	eps	revenue	price	people
0	GOOGL	27.82	87.0	845.0	larry page
1	WALMART	4.61	484.0	65.0	NaN
2	MSFT	-1.00	85.0	64.0	bill gates
3	RIL	NaN	50.0	1023.0	mukeshi ambani
4	TATA	5.60	NaN	NaN	ratan tata

```
In [10]: # Para escrever este arquivo em um .csv
# Observe que os "NaN" ficam vazios e o index é adicionado ao arquivo. Caso se queira salvar sem o index: index=False
df4.to_csv('new_tutorial4.csv')
```

```
In [11]: # Caso se queira exportar somente duas colunas e remover o index:
# Pode-se omitir salvar o cabeçalho usando: header=False
df4.to_csv('new_tutorial4a.csv', index=False, columns=['tickers','revenue'])
```

```
In [12]: # Importando do excel e realizando conversões com funções:

def convert_people_cell(cell):
    if cell == "n.a.":
        return 'Sam Walton'
    return cell

def convert_eps_cell(cell):
    if cell == "not available":
        return None
    return cell

df4 = pd.read_excel('sample_data_tutorial_04.xlsx', 'Sheet2', skiprows=1, converters = {
    'people': convert_people_cell,
    'eps': convert_eps_cell
})
df4
```

Out[12]:

	tickers	eps	revenue	price	people
0	GOOGL	55.64	174	1690	larry page
1	WALMART	9.22	968	130	Sam Walton
2	MSFT	-1.00	170	128	bill gates
3	RIL	NaN	100	2046	mukeshi ambani
4	TATA	11.20	-1	n.a.	ratan tata

```
In [13]: # Para salvar este arquivo no excel (aba=Stocks, ...):
df4.to_excel("new_tutorial4a.xlsx", sheet_name='Stocks', startrow=1, startcol=2)
```

In [14]: *# Caso se queira salvar 2 DataFrame no mesmo arquivo excel em duas abas diferentes:*

```
df_stocks = pd.DataFrame({
    'tickers': ['GOOGLE', 'WALMART', 'MICROSOFT'],
    'price' : [845, 65, 64],
    'pe' : [30.37, 14.26, 30.97],
    'eps' : [27.82, 4.61, 2.12]
})

df_weather = pd.DataFrame({
    'day': ['1/1/2017', '1/2/2017', '1/3/2017'],
    'temp' : [32, 35, 28],
    'event' : ['Rain', 'Sunny', 'Snow']
})
```

In [15]: **with** pd.ExcelWriter('stocks_weather.xlsx') **as** writer:
 df_stocks.to_excel(writer, sheet_name='Stocks')
 df_weather.to_excel(writer, sheet_name='Weather')