

Pandas IO ☆

Arquivo Editar Ver Inserir Ambiente de execução Ferramentas Ajuda Todas as alterações foram salvas

Comentário Compartilhar Configurações Usuário

+ Código + Texto

[1] import pandas as pd

[2] url = 'https://raw.githubusercontent.com/alura-cursos/Pandas/main/superstore_data.csv'

[3] dados_mercado = pd.read_csv(url)

[4] dados_mercado.head()

	Id	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	...	MntFishProducts	MntSweetProducts	MntGoldProds	Nur
0	1826	1970	Graduation	Divorced	84835.0	0	0	6/16/2014	0	189	...	111	189	218	
1	1	1961	Graduation	Single	57091.0	0	0	6/15/2014	0	464	...	7	0	37	
2	10476	1958	Graduation	Married	67267.0	0	1	5/13/2014	0	134	...	15	2	30	
3	1386	1967	Graduation	Together	32474.0	1	1	11/5/2014	0	10	...	0	0	0	
4	5371	1989	Graduation	Single	21474.0	1	0	8/4/2014	0	6	...	11	0	34	

5 rows × 22 columns

[5] url_2 = 'https://raw.githubusercontent.com/alura-cursos/Pandas/main/superstore_data_ponto_virgula.csv'

[6] dados_ponto_virgula = pd.read_csv(url_2)

[7] dados_ponto_virgula.head()

	Id;Year_Birth;Education;Marital_Status;Income;Kidhome;Teenhome;Dt_Customer;Recency;MntWines;MntFruits;MntMeatProducts;MntFishProducts;MntSweetProducts;MntGoldProds
0	
1	
2	
3	
4	

[8] dados_ponto_virgula = pd.read_csv(url_2, sep = ';')

[9] dados_ponto_virgula.head()

	Id	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	...	MntFishProducts	MntSweetProducts	MntGoldProds	Nur
0	1826	1970	Graduation	Divorced	84835.0	0	0	6/16/2014	0	189	...	111	189	218	
1	1	1961	Graduation	Single	57091.0	0	0	6/15/2014	0	464	...	7	0	37	
2	10476	1958	Graduation	Married	67267.0	0	1	5/13/2014	0	134	...	15	2	30	
3	1386	1967	Graduation	Together	32474.0	1	1	11/5/2014	0	10	...	0	0	0	
4	5371	1989	Graduation	Single	21474.0	1	0	8/4/2014	0	6	...	11	0	34	

5 rows × 22 columns

[10] #parâmetro para número de linhas
dados_primeiras_linhas = pd.read_csv(url, nrows=5)

[11] dados_primeiras_linhas

	Id	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	...	MntFishProducts	MntSweetProducts	MntGoldProds	Nur
0	1826	1970	Graduation	Divorced	84835	0	0	6/16/2014	0	189	...	111	189	218	
1	1	1961	Graduation	Single	57091	0	0	6/15/2014	0	464	...	7	0	37	
2	10476	1958	Graduation	Married	67267	0	1	5/13/2014	0	134	...	15	2	30	
3	1386	1967	Graduation	Together	32474	1	1	11/5/2014	0	10	...	0	0	0	
4	5371	1989	Graduation	Single	21474	1	0	8/4/2014	0	6	...	11	0	34	

5 rows × 22 columns

[12] dados_selecao = pd.read_csv(url, usecols=['Id', 'Year_Birth', 'Income'])

[13] dados_selecao.head()

	Id	Year_Birth	Income
0	1826	1970	84835.0
1	1	1961	57091.0
2	10476	1958	67267.0
3	1386	1967	32474.0
4	5371	1989	21474.0

```
[14] dados_selecao = pd.read_csv(url, usecols=[0, 1, 4])
```

```
[15] dados_selecao.head()
```

	Id	Year_Birth	Income
0	1826	1970	84835.0
1	1	1961	57091.0
2	10476	1958	67267.0
3	1386	1967	32474.0
4	5371	1989	21474.0

```
[16] dados_selecao.to_csv('clientes_mercado.csv')
```

```
[17] clientes_mercado = pd.read_csv('/content/clientes_mercado.csv')
```

```
[18] clientes_mercado
```

	Unnamed: 0	Id	Year_Birth	Income
0	0	1826	1970	84835.0
1	1	1	1961	57091.0
2	2	10476	1958	67267.0
3	3	1386	1967	32474.0
4	4	5371	1989	21474.0
...
2235	2235	10142	1976	66476.0
2236	2236	5263	1977	31056.0
2237	2237	22	1976	46310.0
2238	2238	528	1978	65819.0
2239	2239	4070	1969	94871.0

2240 rows × 4 columns

```
[19] dados_selecao.to_csv('clientes_mercado.csv', index=False)
clientes_mercado = pd.read_csv('/content/clientes_mercado.csv')
clientes_mercado
```

	Id	Year_Birth	Income
0	1826	1970	84835.0
1	1	1961	57091.0
2	10476	1958	67267.0
3	1386	1967	32474.0
4	5371	1989	21474.0
...
2235	10142	1976	66476.0
2236	5263	1977	31056.0
2237	22	1976	46310.0
2238	528	1978	65819.0
2239	4070	1969	94871.0

2240 rows × 3 columns

```
[20] import chardet

with open('/content/dados_sus.csv', "rb") as file:
    print(chardet.detect(file.read()))

{'encoding': 'ISO-8859-1', 'confidence': 0.73, 'language': ''}
```

```
[21] gastos_hospitalares_sus = pd.read_csv("/content/dados_sus.csv", sep=';', encoding='ISO-8859-1', skiprows=3, skipfooter=9, engine="python")
```

```
[22] gastos_hospitalares_sus.head()
```

Unidade da Federação	2008/Jan	2008/Fev	2008/Mar	2008/Abr	2008/Mai	2008/Jun	2008/Jul	2008/Ago	2008/Set	...	2020/Jul	2020/Ago	2020/...
0 Rondônia	1388528,39	2931283,42	1541682,52	1525314,96	1645953,84	1406150,68	3065279,01	3231494,61	3117178,63	...	11824687,13	11733307,76	1020198
1 Acre	902416,00	1497206,26	1794028,48	1730469,42	1819443,92	1828496,00	2511754,59	2089107,14	2275708,53	...	3915193,20	3640142,82	339124
2 Amazonas	4735529,42	7118990,57	8196635,49	8259378,42	7831399,90	8477273,62	9368858,72	9352532,70	9363099,35	...	19769460,14	18059931,43	1784101
3 Roraima	657889,53	777939,31	718688,03	839994,39	862347,96	832440,66	996693,09	894271,18	910424,17	...	3015488,30	2826486,18	292804
4 Pará	18864744,11	19553758,20	21937342,70	20842829,69	23249952,88	23240687,56	24002223,56	23341218,03	25172261,32	...	40804126,43	44385715,88	3682024

5 rows × 161 columns

Trabalhando com planilhas

```
[23] url = "https://github.com/alura-cursos/Pandas/blob/main/emissoes_CO2.xlsx?raw=True"
```

```
[24] dados_co2 = pd.read_excel(url)
```

```
[25] dados_co2.head()
```

País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro	Per Capita	grid icon	info icon
0 Afeganistão		AFG	1750	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
1 Afeganistão		AFG	1751	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
2 Afeganistão		AFG	1752	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
3 Afeganistão		AFG	1753	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
4 Afeganistão		AFG	1754	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN		

```
[26] # para saber número de "páginas" ou "abas" da planilha e seus nomes.  
pd.ExcelFile(url).sheet_names
```

```
['emissoes_CO2', 'emissoes_percapita', 'fontes']
```

```
[27] percapita = pd.read_excel(url, sheet_name='emissoes_percapita')
```

```
[28] percapita.head()
```

País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro	Per Capita	grid icon	info icon
0 Afeganistão		AFG	1750	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
1 Afeganistão		AFG	1751	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
2 Afeganistão		AFG	1752	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
3 Afeganistão		AFG	1753	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		
4 Afeganistão		AFG	1754	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		

```
[29] fontes = pd.read_excel(url, sheet_name='fontes')
```

```
[30] fontes.head()
```

País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro	Per Capita	grid icon	info icon
0 Afeganistão		AFG	1750	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]		
1 Afeganistão		AFG	1751	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]		
2 Afeganistão		AFG	1752	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]		
3 Afeganistão		AFG	1753	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]		
4 Afeganistão		AFG	1754	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]		

```
[30]
```

```
[31] intervalo = pd.read_excel(url, sheet_name='emissoes_CO2', usecols='A:D')
```

```
[32] intervalo.head()
```

País	ISO 3166-1	alpha-3	Ano	Total	grid icon	info icon
0 Afeganistão		AFG	1750	0.0		
1 Afeganistão		AFG	1751	0.0		
2 Afeganistão		AFG	1752	0.0		

```
3 Afganistão          AFG 1753  0.0
4 Afganistão          AFG 1754  0.0
```

```
[33] intervalo_2 = pd.read_excel(url, sheet_name='emissoes_C02', usecols='A:D', nrows=10)
```

```
[34] intervalo_2
```

	País	ISO 3166-1	alpha-3	Ano	Total
0	Afganistão	AFG	1750	0	
1	Afganistão	AFG	1751	0	
2	Afganistão	AFG	1752	0	
3	Afganistão	AFG	1753	0	
4	Afganistão	AFG	1754	0	
5	Afganistão	AFG	1755	0	
6	Afganistão	AFG	1756	0	
7	Afganistão	AFG	1757	0	
8	Afganistão	AFG	1758	0	
9	Afganistão	AFG	1759	0	

```
[35] percapita.to_excel('co2_percapita.xlsx', index=False)
```

```
[36] pd.read_excel('/content/co2_percapita.xlsx')
```

	País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro
0	Afganistão	AFG	1750		Nan	Nan	Nan	Nan	Nan	Nan	Nan
1	Afganistão	AFG	1751		Nan	Nan	Nan	Nan	Nan	Nan	Nan
2	Afganistão	AFG	1752		Nan	Nan	Nan	Nan	Nan	Nan	Nan
3	Afganistão	AFG	1753		Nan	Nan	Nan	Nan	Nan	Nan	Nan
4	Afganistão	AFG	1754		Nan	Nan	Nan	Nan	Nan	Nan	Nan
...
63099	Global	WLD	2017	4749682.0	1908857.0	1610910.0	940144.0	198416.0	51579.0	39776.0	
63100	Global	WLD	2018	4792753.0	1919213.0	1596350.0	979965.0	204225.0	53634.0	39366.0	
63101	Global	WLD	2019	4775633.0	1896468.0	1589920.0	984878.0	208309.0	56569.0	39490.0	
63102	Global	WLD	2020	4497423.0	1807760.0	1427353.0	963695.0	208844.0	51981.0	37789.0	
63103	Global	WLD	2021	4693699.0	1893923.0	1496614.0	1001585.0	211472.0	52663.0	37443.0	

63104 rows × 10 columns

```
[37] 'https://docs.google.com/spreadsheets/d/1Bq9pL2A3qPgCcUqP2gby9CyenS10YZ1XwbHoeQcHSWc/edit?usp=sharing'
```

'https://docs.google.com/spreadsheets/d/1Bq9pL2A3qPgCcUqP2gby9CyenS10YZ1XwbHoeQcHSWc/edit?usp=sharing'

```
[38] sheet_id = '1Bq9pL2A3qPgCcUqP2gby9CyenS10YZ1XwbHoeQcHSWc'
```

```
[39] url = f'https://docs.google.com/spreadsheets/d/{sheet_id}/gviz/tq?tqx=out:csv&sheet='
```

```
[40] dados_co2_sheets = pd.read_csv(url)
```

```
[41] dados_co2_sheets.head()
```

	País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro	Per Capita
0	Afganistão	AFG	1750	0	Nan	Nan	Nan	Nan	Nan	Nan	Nan	Nan
1	Afganistão	AFG	1751	0	Nan	Nan	Nan	Nan	Nan	Nan	Nan	Nan
2	Afganistão	AFG	1752	0	Nan	Nan	Nan	Nan	Nan	Nan	Nan	Nan
3	Afganistão	AFG	1753	0	Nan	Nan	Nan	Nan	Nan	Nan	Nan	Nan
4	Afganistão	AFG	1754	0	Nan	Nan	Nan	Nan	Nan	Nan	Nan	Nan

```
[42] # agora escolhendo a planilha do arquivo
sheet_name = 'emissoes_percapita'
url_percapita = f'https://docs.google.com/spreadsheets/d/{sheet_id}/gviz/tq?tqx=out:csv&sheet={sheet_name}'
```

```
[43] percapita_sheets = pd.read_csv(url_percapita)
percapita_sheets
```

	País	ISO 3166-1	alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro
0	Afganistão	AFG	1750		Nan	Nan	Nan	Nan	Nan	Nan	Nan

1	Afganistão	AFG	1751	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	Afganistão	AFG	1752	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	Afganistão	AFG	1753	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	Afganistão	AFG	1754	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...
63099	Global	WLD	2017	4.749.682	1.908.857	1.610.910	940.144	198.416	51.579	39.776	
63100	Global	WLD	2018	4.792.753	1.919.213	1.596.350	979.965	204.225	53.634	39.366	
63101	Global	WLD	2019	4.775.633	1.896.468	1.589.920	984.878	208.309	56.569	39.490	
63102	Global	WLD	2020	4.497.423	1.807.760	1.427.353	963.695	208.844	51.981	37.789	
63103	Global	WLD	2021	4.693.699	1.893.923	1.496.614	1.001.585	211.472	52.663	37.443	

63104 rows × 10 columns

```
[44] sheet_name = 'fontes'
    url_fontes = f'https://docs.google.com/spreadsheets/d/{sheet_id}/gviz/tq?tqx=out:csv&sheet={sheet_name}'
```

```
✓ [45] fontes_sheets = pd.read_csv(url_fontes)
      fontes_sheets
```

	País	ISO 3166-1 alpha-3	Ano	Total	Carvão	Óleo	Gás	Cimento	Queima	Outro	Per Capita
0	Afeganistão	AFG	1750	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]
1	Afeganistão	AFG	1751	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]
2	Afeganistão	AFG	1752	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]
3	Afeganistão	AFG	1753	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]
4	Afeganistão	AFG	1754	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]	[NONE]
...
63099	Global	WLD	2017	CDIAC 2022, BP, and Sum of countries	CDIAC 2022 and BP	CDIAC 2022 and BP	CDIAC 2022 and BP	Andrew cement	CDIAC 2022 and GCP	[NONE]	CDIAC 2022, BP, Sum of countries, and UN popul..
63100	Global	WLD	2018	CDIAC 2022, BP, and Sum of countries	CDIAC 2022 and BP	CDIAC 2022 and BP	CDIAC 2022 and BP	Andrew cement	CDIAC 2022 and GCP	[NONE]	CDIAC 2022, BP, Sum of countries, and UN popul..
63101	Global	WLD	2019	CDIAC 2022, BP, and Sum of countries	CDIAC 2022 and BP	CDIAC 2022 and BP	CDIAC 2022 and BP	Andrew cement	CDIAC 2022 and GCP	[NONE]	CDIAC 2022, BP, Sum of countries, and UN popul..
63102	Global	WLD	2020	CDIAC 2022, BP, and Sum of countries	CDIAC 2022 and BP	CDIAC 2022 and BP	CDIAC 2022 and BP	Andrew cement	CDIAC 2022 and GCP	[NONE]	CDIAC 2022, BP, Sum of countries, and UN popul..
63103	Global	WLD	2021	CDIAC 2022, BP, and Sum of countries	CDIAC 2022 and BP	CDIAC 2022 and BP	CDIAC 2022 and BP	Andrew cement	CDIAC 2022 and GCP	[NONE]	CDIAC 2022, BP, Sum of countries, and UN popul..

63104 rows x 11 columns

▼ Trabalhando com Json

```
✓ 0s ➔ dados_pacientes = pd.read_json("/content/pacientes.json")
dados pacientes
```

1000 rows × 19 columns

```
✓ [47] # arquivo json com informações aninhadas.  
os dados_pacientes_2 = pd.read_json("/content/pacientes_2.json")  
dados_pacientes_2
```

```
1 Principais Indicadores de Doenca Cardiaca 2020 {ID: '02', 'Faixa_etaria': '60-64+', 'Sexo_b...}
```

```
2 Principais Indicadores de Doenca Cardiaca 2020 {ID: '03', 'Faixa_etaria': '65-69', 'Sexo_b...
```

```
[48] df_normalizado = pd.json_normalize(dados_pacientes_2['Pacientes'])
```

```
[49] df_normalizado
```

ID	Faixa_etaria	Sexo_biológico	Raça	IMC	Fumante	Consumo_alcool	Saude_física	Saude_mental	Dificuldade_caminhar	Atividade_física	Saude_geral	Horas_s...
0	01	55-59	Feminino	Branca	16.60	Sim	Nao	3	30	Nao	Sim	Muito boa
1	02	80 ou +	Feminino	Branca	20.34	Nao	Nao	0	0	Nao	Sim	Muito boa
2	03	65-69	Masculino	Branca	26.58	Sim	Nao	20	30	Nao	Sim	Muito boa

```
[50] df_normalizado.to_json('historico_pacientes_normalizado.json')
```

```
[51] pd.read_json('/content/historico_pacientes_normalizado.json')
```

ID	Faixa_etaria	Sexo_biológico	Raça	IMC	Fumante	Consumo_alcool	Saude_física	Saude_mental	Dificuldade_caminhar	Atividade_física	Saude_geral	Horas_s...
0	1	55-59	Feminino	Branca	16.60	Sim	Nao	3	30	Nao	Sim	Muito boa
1	2	80 ou +	Feminino	Branca	20.34	Nao	Nao	0	0	Nao	Sim	Muito boa
2	3	65-69	Masculino	Branca	26.58	Sim	Nao	20	30	Nao	Sim	Muito boa

```
[52] # módulos requests e json do python para extrair infos de APIs
```

```
import requests
```

```
import json
```

```
dados_frutas = requests.get('https://fruityvice.com/api/fruit/all')
```

```
resultado = json.loads(dados_frutas.text)
```

```
api_frutas = pd.DataFrame(resultado)
```

```
api_frutas
```

10	Passionfruit	70	Passifloraceae	Malpighiales	Passiflora	{'calories': 97, 'fat': 0.7, 'sugar': 11.2, 'c...
14	Plum	71	Rosaceae	Rosales	Prunus	{'calories': 46, 'fat': 0.28, 'sugar': 9.92, ...}
15	Orange	2	Rutaceae	Sapindales	Citrus	{'calories': 43, 'fat': 0.2, 'sugar': 8.2, 'ca...
16	GreenApple	72	Rosaceae	Rosales	Malus	{'calories': 21, 'fat': 0.1, 'sugar': 6.4, 'ca...
17	Raspberry	23	Rosaceae	Rosales	Rubus	{'calories': 53, 'fat': 0.7, 'sugar': 4.4, 'ca...
18	Watermelon	25	Cucurbitaceae	Cucurbitales	Citrullus	{'calories': 30, 'fat': 0.2, 'sugar': 6.0, 'ca...
19	Lemon	26	Rutaceae	Sapindales	Citrus	{'calories': 29, 'fat': 0.3, 'sugar': 2.5, 'ca...
20	Mango	27	Anacardiaceae	Sapindales	Mangifera	{'calories': 60, 'fat': 0.38, 'sugar': 13.7, ...}
21	Blueberry	33	Rosaceae	Rosales	Fragaria	{'calories': 29, 'fat': 0.4, 'sugar': 5.4, 'ca...
22	Apple	6	Rosaceae	Rosales	Malus	{'calories': 52, 'fat': 0.4, 'sugar': 10.3, 'ca...
23	Guava	37	Myrtaceae	Myrtales	Psidium	{'calories': 68, 'fat': 1.0, 'sugar': 9.0, 'ca...
24	Apricot	35	Rosaceae	Rosales	Prunus	{'calories': 15, 'fat': 0.1, 'sugar': 3.2, 'ca...
25	Papaya	42	Caricaceae	Caricae	Carica	{'calories': 43, 'fat': 0.4, 'sugar': 1.0, 'ca...
26	Melon	41	Cucurbitaceae	Cucurbitales	Cucumis	{'calories': 34, 'fat': 0.0, 'sugar': 8.0, 'ca...
27	Tangerine	77	Rutaceae	Sapindales	Citrus	{'calories': 45, 'fat': 0.4, 'sugar': 9.1, 'ca...
28	Pitahaya	78	Cactaceae	Caryophyllales	Cactaceae	{'calories': 36, 'fat': 0.4, 'sugar': 3.0, 'ca...
29	Lime	44	Rutaceae	Sapindales	Citrus	{'calories': 25, 'fat': 0.1, 'sugar': 1.7, 'ca...
30	Pomegranate	79	Lythraceae	Myrtales	Punica	{'calories': 83, 'fat': 1.2, 'sugar': 13.7, 'ca...
31	Dragonfruit	80	Cactaceae	Caryophyllales	Selenicereus	{'calories': 60, 'fat': 1.5, 'sugar': 8.0, 'ca...
32	Grape	81	Vitaceae	Vitales	Vitis	{'calories': 69, 'fat': 0.16, 'sugar': 16.0, 'ca...
33	Morus	82	Moraceae	Rosales	Morus	{'calories': 43, 'fat': 0.39, 'sugar': 8.1, 'ca...
34	Feijoia	76	Myrtaceae	Myrtoideae	Sellowiana	{'calories': 44, 'fat': 0.4, 'sugar': 3.0, 'ca...
35	Avocado	84	Lauraceae	Laurales	Persea	{'calories': 160, 'fat': 14.66, 'sugar': 0.66, ...}
36	Kiwifruit	85	Actinidiaceae	Ericales	Actinidia	{'calories': 61, 'fat': 0.5, 'sugar': 8.9, 'ca...
37	Cranberry	87	Ericaceae	Ericales	Vaccinium	{'calories': 46, 'fat': 0.1, 'sugar': 4.0, 'ca...
38	Cherry	9	Rosaceae	Rosales	Prunus	{'calories': 50, 'fat': 0.3, 'sugar': 8.0, 'ca...
39	Peach	86	Rosaceae	Rosales	Prunus	{'calories': 39, 'fat': 0.25, 'sugar': 8.4, 'ca...
40	Jackfruit	94	Moraceae	Rosales	Artocarpus	{'calories': 95, 'fat': 0.0, 'sugar': 19.1, 'ca...
41	Horned Melon	95	Cucurbitaceae	Cucurbitales	Cucumis	{'calories': 44, 'fat': 1.26, 'sugar': 0.5, 'ca...
42	Hazelnut	96	Betulaceae	Fagales	Corylus	{'calories': 628, 'fat': 61.0, 'sugar': 4.3, 'ca...
43	Pomelo	98	Rutaceae	Sapindales	Citrus	{'calories': 37, 'fat': 0.0, 'sugar': 8.5, 'ca...

```

✓ [53] # normalizando a coluna "nutritions"
nutritions_normalizado = pd.json_normalize(api_frutas['nutritions'])
nutritions_normalizado
   id    sfr  0.70  11.20    22.40    2.20
  13      97  0.70  11.20    22.40    2.20
  14      46  0.28  9.92   11.40    0.70
  15      43  0.20  8.20    8.30    1.00
  16      21  0.10  6.40    3.10    0.40
  17      53  0.70  4.40   12.00    1.20
  18      30  0.20  6.00    8.00    0.60
  19      29  0.30  2.50    9.00    1.10
  20      60  0.38 13.70   15.00    0.82
  21      29  0.40  5.40    5.50    0.00
  22      52  0.40 10.30   11.40    0.30
  23      68  1.00  9.00   14.00    2.60
  24      15  0.10  3.20    3.90    0.50
  25      43  0.40  1.00   11.00    0.00
  26      34  0.00  8.00    8.00    0.00
  27      45  0.40  9.10    8.30    0.00
  28      36  0.40  3.00    7.00    1.00
  29      25  0.10  1.70    8.40    0.30
  30      83  1.20 13.70   18.70    1.70
  31      60  1.50  8.00    9.00    9.00
  32      69  0.16 16.00   18.10    0.72
  33      43  0.39  8.10    9.80    1.44
  34      44  0.40  3.00    8.00    0.60
  35     160  14.66  0.66    8.53    2.00
  36      61  0.50  8.90   14.60    1.14
  37      46  0.10  4.00   12.20    0.40
  38      50  0.30  8.00   12.00    1.00
  39      39  0.25  8.40    9.50    0.90
  40      95  0.00 19.10   23.20    1.72
  41      44  1.26  0.50    7.56    1.78
  42     628  61.00  4.30   17.00   15.00
  43      37  0.00  8.50    9.67    0.82
  44      73  0.58 16.11   17.91    0.41

```

Manipulando HMTL

```

✓ [54] dados_html = pd.read_html("/content/filmes_wikipedia.html")

✓ [55] dados_html
[4]:
  4                               2002
  5                               2003
  6                               2004
  7                               2005
  8                               2005
  9                               2006
 10                              2006
 11                              2007
 12                              2008
 13 .mw-parser-output .hlist dl.mw-parser-output ...
          1
  0           100 Movies
  1           100 Stars
  2           100 Laughs
  3           100 Thrills
  4           100 Passions
  5           100 Heroes & Villains
  6           100 Songs
  7           100 Movie Quotes
  8           25 Scores
  9           100 Cheers
 10          25 Musicals
 11          100 Movies (Updated)
 12          AFI's 10 Top 10
 13 .mw-parser-output .hlist dl.mw-parser-output ... ,
          Film Release year           Director \

```

```

0           Citizen Kane      1941      Orson Welles
1           Casablanca        1942      Michael Curtiz
2           The Godfather     1972  Francis Ford Coppola
3           Gone with the Wind 1939      Victor Fleming
4           Lawrence of Arabia 1962      David Lean
...
95          ...                ...
95          The Searchers     1956      John Ford
96          Bringing Up Baby 1938      Howard Hawks
97          Unforgiven        1992  Clint Eastwood
98 Guess Who's Coming to Dinner 1967  Stanley Kramer
99           Yankee Doodle Dandy 1942      Michael Curtiz

```

```

Production companies  Rank
0           RKO Radio Pictures  1
1           Warner Bros. Pictures  2
2 Paramount Pictures, Alfran Productions  3
3           Selznick International Pictures  4
4           Horizon Pictures  5
...
95          ...                ...
95          C. V. Whitney Pictures  96
96          RKO Radio Pictures  97
97          The Malpaso Company  98
98          Columbia Pictures  99
99          Warner Bros. Pictures  100

```

```

[100 rows x 5 columns],
hidevteAFI's 100 Years...100 Movies (1998) \
0 Citizen Kane (1941) Casablanca (1942) The Godf...
hidevteAFI's 100 Years...100 Movies (1998).1
0 Citizen Kane (1941) Casablanca (1942) The Godf... ]

```

✓ [56] type(dados_html)

list

✓ [57] len(dados_html)

3

✓ [58] top_filmes = dados_html[1]

✓ [59] top_filmes

	Film	Release year	Director	Production companies	Rank		
0	Citizen Kane	1941	Orson Welles	RKO Radio Pictures	1		
1	Casablanca	1942	Michael Curtiz	Warner Bros. Pictures	2		
2	The Godfather	1972	Francis Ford Coppola	Paramount Pictures, Alfran Productions	3		
3	Gone with the Wind	1939	Victor Fleming	Selznick International Pictures	4		
4	Lawrence of Arabia	1962	David Lean	Horizon Pictures	5		
...		
95	The Searchers	1956	John Ford	C. V. Whitney Pictures	96		
96	Bringing Up Baby	1938	Howard Hawks	RKO Radio Pictures	97		
97	Unforgiven	1992	Clint Eastwood	The Malpaso Company	98		
98	Guess Who's Coming to Dinner	1967	Stanley Kramer	Columbia Pictures	99		
99	Yankee Doodle Dandy	1942	Michael Curtiz	Warner Bros. Pictures	100		

100 rows x 5 columns

✓ [60] top_filmes.to_html("top_filmes.html")

✓ [61] top_filmes.to_csv("top_filmes_1998.csv", index=False)

✓ [62] pd.read_csv('/content/top_filmes_1998.csv')

	Film	Release year	Director	Production companies	Rank		
0	Citizen Kane	1941	Orson Welles	RKO Radio Pictures	1		
1	Casablanca	1942	Michael Curtiz	Warner Bros. Pictures	2		
2	The Godfather	1972	Francis Ford Coppola	Paramount Pictures, Alfran Productions	3		
3	Gone with the Wind	1939	Victor Fleming	Selznick International Pictures	4		
4	Lawrence of Arabia	1962	David Lean	Horizon Pictures	5		
...		
95	The Searchers	1956	John Ford	C. V. Whitney Pictures	96		
96	Bringing Up Baby	1938	Howard Hawks	RKO Radio Pictures	97		
97	Unforgiven	1992	Clint Eastwood	The Malpaso Company	98		
98	Guess Who's Coming to Dinner	1967	Stanley Kramer	Columbia Pictures	99		
99	Yankee Doodle Dandy	1942	Michael Curtiz	Warner Bros. Pictures	100		

100 rows × 5 columns

▼ Trabalhando com XML

```
[63] dados_imdb = pd.read_xml("/content/imdb_top_1000.xml")
```

```
[64] dados_imdb.head(3)
```

index		Poster_Link	Series_Title	Released_Year	Certificate	Runtime	Genre	IMDB_Rating	Overview	Meta_score	Director	Star1	St...
0	0	https://m.media-amazon.com/images/M/MV5BMDFK...	The Shawshank Redemption	1994	A	142 min	Drama	9.3	Two imprisoned men bond over a number of years...	80.0	Frank Darabont	Tim Robbins	More Fre...
1	1	https://m.media-amazon.com/images/M/MV5BM2MyNj...	The Godfather	1972	A	175 min	Crime, Drama	9.2	An organized crime dynasty's aging patriarch ...	100.0	Francis Ford Coppola	Marlon Brando	Pat Pac...
2	2	https://m.media-amazon.com/images/M/MV5BMTMxNT...	The Dark Knight	2008	UA	152 min	Action, Crime, Drama	9.0	When the menace known as the Joker wreaks havo...	84.0	Christopher Nolan	Christian Bale	Heath Led...

```
[65] dados_imdb.to_xml('dados_imdb.xml')
```

▼ Trabalhando com Banco de Dados

```
[66] !pip install --upgrade 'sqlalchemy<2.0'
```

```
Requirement already satisfied: sqlalchemy<2.0 in /usr/local/lib/python3.10/dist-packages (1.4.50)
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from sqlalchemy<2.0) (3.0.0)
```

```
[67] import sqlalchemy
     from sqlalchemy import create_engine, MetaData, Table, inspect
```

```
[68] engine = create_engine("sqlite:///memory:")
```

```
[69] banco = "/content/clientes_banco.csv"
     dados = pd.read_csv(banco)
```

```
[70] dados.head()
```

ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	Moradia	
0	5008804	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento alugado
1	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento alugado
2	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1	Casa/apartamento próprio
3	5008808	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0	Casa/apartamento próprio
4	5008809	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0	Casa/apartamento próprio

```
[71] dados.to_sql('clientes', engine, index=False)
```

```
438463
```

```
[72] inspector = inspect(engine)
```

```
[73] print(inspector.get_table_names())
```

```
['clientes']
```

```
[74] query = 'SELECT * FROM clientes WHERE Categoria_de_renda="Empregado"
```

```
[75] empregados = pd.read_sql(query, engine)
```

✓ [76] empregados

	ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	M
0	5008804	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento
1	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento
2	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1	Casa/apartamento
3	5008815	46	Ensino superior	Casado	2	Empregado	Contabilidade	2	270000.0	1	Casa/apartamento
4	5112956	46	Ensino superior	Casado	2	Empregado	Contabilidade	2	270000.0	1	Casa/apartamento
...
226054	6837905	43	Ensino médio	Casado	3	Empregado	Outro	7	355050.0	1	Casa/apartamento
226055	6837906	43	Ensino médio	Casado	3	Empregado	Outro	7	355050.0	1	Casa/apartamento
226056	6839936	34	Ensino médio	Casado	3	Empregado	Construção Civil	5	135000.0	1	Casa/apartamento
226057	6840222	43	Ensino médio	Solteiro	1	Empregado	Construção Civil	8	103500.0	0	Casa/apartamento
226058	6842885	51	Ensino médio	Casado	2	Empregado	Vendas	3	121500.0	0	Casa/apartamento

226059 rows × 11 columns

✓ [77] empregados.to_sql('empregados', con=engine, index=False)

226059

✓ [78] pd.read_sql_table('empregados', engine)

	ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	M
0	5008804	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento
1	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1	Apartamento
2	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1	Casa/apartamento
3	5008815	46	Ensino superior	Casado	2	Empregado	Contabilidade	2	270000.0	1	Casa/apartamento
4	5112956	46	Ensino superior	Casado	2	Empregado	Contabilidade	2	270000.0	1	Casa/apartamento
...
226054	6837905	43	Ensino médio	Casado	3	Empregado	Outro	7	355050.0	1	Casa/apartamento
226055	6837906	43	Ensino médio	Casado	3	Empregado	Outro	7	355050.0	1	Casa/apartamento
226056	6839936	34	Ensino médio	Casado	3	Empregado	Construção Civil	5	135000.0	1	Casa/apartamento
226057	6840222	43	Ensino médio	Solteiro	1	Empregado	Construção Civil	8	103500.0	0	Casa/apartamento
226058	6842885	51	Ensino médio	Casado	2	Empregado	Vendas	3	121500.0	0	Casa/apartamento

226059 rows × 11 columns

✓ [79] pd.read_sql_table('empregados', engine, columns=['ID_Cliente', 'Grau_escolaridade', 'Rendimento_anual'])

ID_Cliente	Grau_escolaridade	Rendimento_anual	
0	5008804	Ensino superior	427500.0
1	5008805	Ensino superior	427500.0
2	5008806	Ensino médio	112500.0
3	5008815	Ensino superior	270000.0
4	5112956	Ensino superior	270000.0
...	
226054	6837905	Ensino médio	355050.0
226055	6837906	Ensino médio	355050.0
226056	6839936	Ensino médio	135000.0

```
226057 6840222 Ensino médio 103500.0
226058 6842885 Ensino médio 121500.0
```

226059 rows x 3 columns

```
[80] query = 'SELECT * FROM clientes'
```

```
[81] pd.read_sql(query, engine)
```

ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	Mor
0	5008804	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1 Apartan
1	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1 Apartan
2	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1 Casa/apartan
3	5008808	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0 Casa/apartan
4	5008809	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0 Casa/apartan
...
438458	6840104	62	Ensino médio	Divorciado	1	Pensionista	Outro	0	135000.0	0 Casa/apartan
438459	6840222	43	Ensino médio	Solteiro	1	Empregado	Construção Civil	8	103500.0	0 Casa/apartan
438460	6841878	22	Ensino superior	Solteiro	1	Associado comercial	Vendas	1	54000.0	0 Mora com os
438461	6842765	59	Ensino médio	Casado	2	Pensionista	Outro	0	72000.0	0 Casa/apartan
438462	6842885	51	Ensino médio	Casado	2	Empregado	Vendas	3	121500.0	0 Casa/apartan

438463 rows x 11 columns

```
[82] #deletando cliente
```

```
query = 'DELETE FROM clientes WHERE ID_Cliente=5008804'
with engine.connect() as conn:
    conn.execute(query)
```

```
[83] pd.read_sql_table('clientes', engine)
```

ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	Mor
0	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1 Apartan
1	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1 Casa/apartan
2	5008808	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0 Casa/apartan
3	5008809	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0 Casa/apartan
4	5008810	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0 Casa/apartan
...
438457	6840104	62	Ensino médio	Divorciado	1	Pensionista	Outro	0	135000.0	0 Casa/apartan
438458	6840222	43	Ensino médio	Solteiro	1	Empregado	Construção Civil	8	103500.0	0 Casa/apartan
438459	6841878	22	Ensino superior	Solteiro	1	Associado comercial	Vendas	1	54000.0	0 Mora com os
438460	6842765	59	Ensino médio	Casado	2	Pensionista	Outro	0	72000.0	0 Casa/apartan
438461	6842885	51	Ensino médio	Casado	2	Empregado	Vendas	3	121500.0	0 Casa/apartan

438462 rows x 11 columns

```
[85] query = 'UPDATE clientes SET Grau_escolaridade="Ensino superior" WHERE ID_Cliente=5008808'
with engine.connect() as conn:
    conn.execute(query)
```

```
[86] pd.read_sql_table('clientes', engine)
```

ID_Cliente	Idade	Grau_escolaridade	Estado_civil	Tamanho_familia	Categoria_de_renda	Ocupacao	Anos_empregado	Rendimento_anual	Tem_carro	Mor
0	5008805	32	Ensino superior	União-estável	2	Empregado	Outro	12	427500.0	1 Apartan

1	5008806	58	Ensino médio	Casado	2	Empregado	Segurança	3	112500.0	1	Casa/apartamento pr
2	5008808	52	Ensino superior	Solteiro	1	Associado comercial	Vendas	8	270000.0	0	Casa/apartamento pr
3	5008809	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0	Casa/apartamento pr
4	5008810	52	Ensino médio	Solteiro	1	Associado comercial	Vendas	8	270000.0	0	Casa/apartamento pr
...
438457	6840104	62	Ensino médio	Divorciado	1	Pensionista	Outro	0	135000.0	0	Casa/apartamento pr
438458	6840222	43	Ensino médio	Solteiro	1	Empregado	Construção Civil	8	103500.0	0	Casa/apartamento pr
438459	6841878	22	Ensino superior	Solteiro	1	Associado comercial	Vendas	1	54000.0	0	Mora com os pais
438460	6842765	59	Ensino médio	Casado	2	Pensionista	Outro	0	72000.0	0	Casa/apartamento pr
438461	6842885	51	Ensino médio	Casado	2	Empregado	Vendas	3	121500.0	0	Casa/apartamento pr

438462 rows × 11 columns

[]

Produtos pagos do Colab - Cancelar contratos

✓ 3s conclusão: 08:34

