

Educação Profissional Paulista

Técnico em
**Ciência de
Dados**

Bibliotecas: Pandas, NumPy, SciPy, Matplotlib e Seaborn

Pandas: combinando *DataFrames*

Aula 1

Código da aula: [DADOS]ANO1C2B4S26A1

**Bibliotecas: Pandas,
NumPy, SciPy,
Matplotlib e Seaborn**

Mapa da Unidade 5 Componente 3

semana

25

Pandas:
transformando
DataFrame

Você está aqui!

Pandas: combinando
DataFrames

semana

26

semana

23

Pandas: acesso
e seleção

semana

24

Pandas: ler e
escrever

**Bibliotecas: Pandas,
NumPy, SciPy,
Matplotlib e Seaborn**

Mapa da Unidade 5 Componente 3

Você está aqui!

**Pandas: combinando
DataFrames**

Aula 1

Código da aula: [DADOS]ANO1C2B4S26A1

26



Objetivos da Aula

- Filtrar um DataFrame usando a biblioteca Pandas do Python.



Recursos Didáticos

- Recurso audiovisual para exibição de vídeos e imagens;
- Acesso ao laboratório de informática e/ou internet;
- Software Anaconda/Jupyter Notebook instalado ou similar.



Duração da Aula

50 minutos.



Competências Técnicas

- Ser proficiente em linguagens de programação para manipular e analisar grandes conjuntos de dados;
- Usar técnicas para explorar e analisar dados, aplicar modelos estatísticos, identificar padrões, realizar inferências e tomar decisões baseadas em evidências.



Competências Socioemocionais

- Colaborar efetivamente com outros profissionais, como cientistas de dados e engenheiros de dados;
- Trabalhar em equipes multifuncionais, colaborando com colegas, gestores e clientes.



Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24	Q25	Q26	Q27	Q28	Q29	Q30	Q31	Q32	Q33	Q34	Q35	Q36	Q37	Q38	Q39	Q40	Q41	Q42	Q43	Q44	Q45	Q46	Q47	Q48	Q49	Q50	Q51	Q52	Q53	Q54	Q55	Q56	Q57	Q58	Q59	Q60	Q61	Q62	Q63	Q64	Q65	Q66	Q67	Q68	Q69	Q70	Q71	Q72	Q73	Q74	Q75	Q76	Q77	Q78	Q79	Q80	Q81	Q82	Q83	Q84	Q85	Q86	Q87	Q88	Q89	Q90	Q91	Q92	Q93	Q94	Q95	Q96	Q97	Q98	Q99	Q100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65																																			

Primeiras ideias

No que a mulher está pensando?

É possível duas tabelas distintas, mas que contêm uma coluna em comum, ou seja, estão relacionadas?

É possível juntar as informações de duas tabelas em uma única?

Como combinar duas tabelas? Juntar na horizontal? Na vertical? Ou relacionada a uma chave?

Ponto de partida

	Nome	Turma	Nasc	Escola	C1	C2	C3	C4	C5	media	media_ponderada
0	Micael Miguel Ehenvic Puzuho de Oliveira	Turma_5	30/01/2007	Escola 2	10.0	7.0	9.5	9.5	10.0	9.000	8.85
1	Janaína Mupru Froiri	Turma_3	18/09/2008	Escola 1	10.0	3.5	10.0	10.0	5.0	8.375	8.05
2	Álvaro Savo da Silva	Turma_4	06/07/2006	Escola 1	10.0	7.5	7.0	7.5	7.5	8.000	7.85
3	Marlon Rosatto Neto	Turma_1	16/10/2006	Escola 2	7.0	5.0	10.0	8.0	0.5	7.500	7.50
4	Sandra Inetirn	Turma_3	08/07/2006	Escola 1	4.0	3.0	6.5	7.5	7.5	5.250	5.15
5	Charlene Sandra Klein de Simões	Turma_3	02/07/2006	Escola 1	10.0	1.0	10.0	8.5	0.0	7.375	7.00
6	Gilberto Martinho Riga	Turma_2	05/05/2007	Escola 2	6.0	9.5	9.5	4.0	3.5	7.250	7.70
7	Gilson de Albuquerque	Turma_1	15/01/2007	Escola 3	8.0	9.5	8.0	10.0	2.0	8.875	8.85
8	Fred Jair dos Santos	Turma_4	03/08/2008	Escola 1	6.0	4.5	9.5	5.5	10.0	6.375	6.50
9	Damião Gowa de Soares	Turma_5	28/02/2009	Escola 1	2.0	8.5	6.5	7.5	4.0	6.125	6.40
10	Filipe Anakin de Padilha	Turma_1	01/12/2008	Escola 1	5.0	8.5	8.5	10.0	7.5	8.000	8.10
11	Guiomar Mourão	Turma_2	04/08/2008	Escola 2	8.5	3.0	10.0	5.5	10.0	6.750	6.70
12	Graziele Minu	Turma_3	23/11/2008	Escola 1	7.0	4.0	10.0	10.0	10.0	7.750	7.60
13	Oliver Doma Teles	Turma_2	09/01/2009	Escola 2	8.0	6.5	7.0	10.0	0.0	7.875	7.65

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

A partir da planilha, você consegue responder às seguintes questões?

1. Se a média final é 7,0, quais alunos passaram?
2. Como filtrar os alunos aprovados?
3. Como separar os alunos apenas da Turma 5 da Escola 2?

Construindo o conceito

Filtrando linhas com base nos valores das colunas

Filtrando linhas com valor específico em uma coluna

Para selecionar linhas que contenham um valor específico em uma coluna, você pode usar a indexação booleana.

```
import pandas as pd

data = {
    "Date": ["April-10", "April-11", "April-12", "April-13", "April-14", "April-16"],
    "Sales": [200, 300, 400, 200, 300, 300],
    "Price": [3, 1, 2, 4, 3, 2]
}

df = pd.DataFrame(data)
df
```

	Date	Sales	Price
0	April-10	200	3
1	April-11	300	1
2	April-12	400	2
3	April-13	200	4
4	April-14	300	3
5	April-16	300	2

```
# Filtrando linhas com Sales igual a 300
df_mask = df["Sales"] == 300
df_filtrado = df[df_mask]

df_filtrado
```

	Date	Sales	Price
1	April-11	300	1
4	April-14	300	3
5	April-16	300	2

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo o conceito

Filtrando linhas com base nos valores das colunas

Uma outra forma é usando a máscara diretamente no DataFrame, e uma outra forma é usando o `.loc()`:

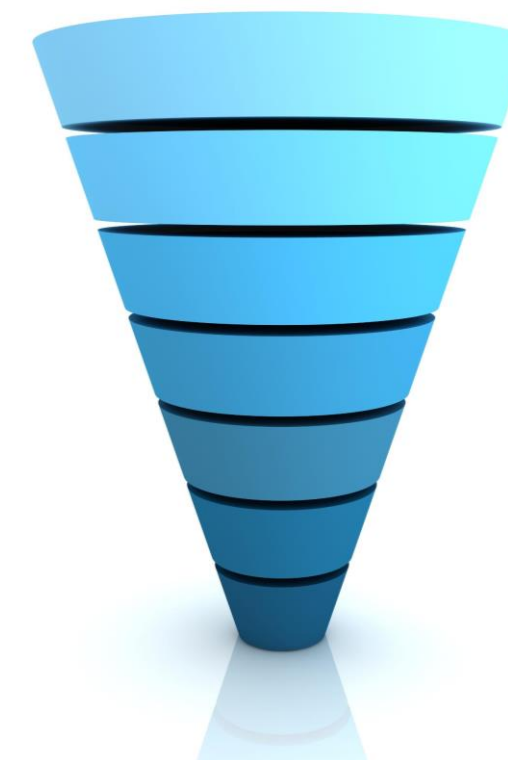
```
df.loc[df["Sales"] == 300]
```

	Date	Sales	Price
1	April-11	300	1
4	April-14	300	3
5	April-16	300	2

```
df[df["Sales"] == 300]
```

	Date	Sales	Price
1	April-11	300	1
4	April-14	300	3
5	April-16	300	2

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



Elaborado especialmente para o curso com
apoio da ferramenta Microsoft Copilot.

Construindo o **conceito**

Filtrando colunas

Para filtrar colunas, você pode usar a função **.loc[]**. Por exemplo, para selecionar apenas as colunas "Date" e "Sales":

```
colunas_selecionadas = df.loc[:, ["Date", "Sales"]]  
colunas_selecionadas
```

	Date	Sales
0	April-10	200
1	April-11	300
2	April-12	400
3	April-13	200
4	April-14	300
5	April-16	300

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



Elaborado especialmente para o curso com apoio da ferramenta Microsoft Copilot.

Construindo o conceito

Filtrando linhas e colunas

```
df.loc[df["Sales"] == 300, ["Date", "Sales"]]
```

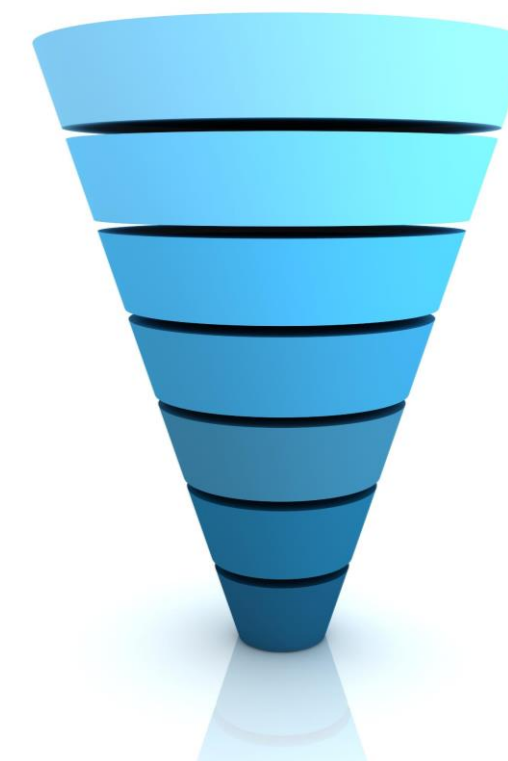
	Date	Sales
1	April-11	300
4	April-14	300
5	April-16	300

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

df: Refere-se ao DataFrame (estrutura de dados tabular) que estamos manipulando.

.loc[]: É um método do Pandas usado para acessar um grupo de linhas e colunas por rótulos ou uma matriz booleana.

[df["Sales"] == 300]: É uma condição booleana aplicada à coluna "Sales". Ela cria uma máscara booleana, onde cada linha é marcada como True se o valor da coluna "Sales" for igual a 300 e False, caso contrário.



Elaborado especialmente para o curso com apoio da ferramenta Microsoft Copilot.

Construindo o **conceito**

Filtrando linhas e colunas

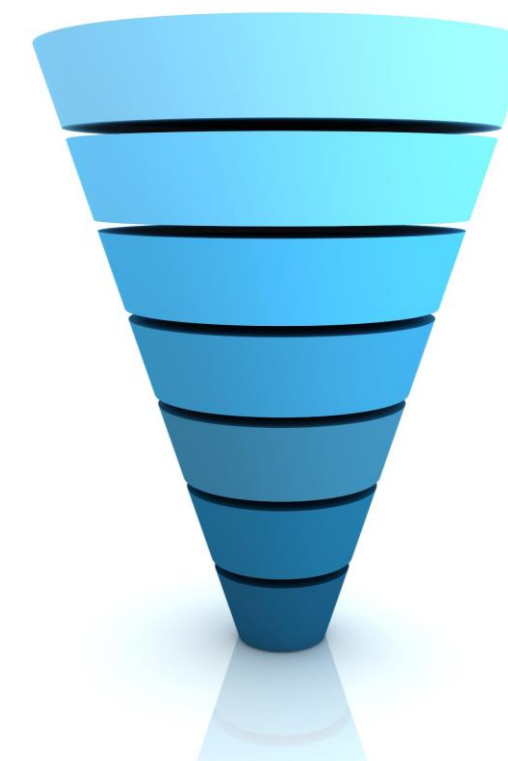
```
df.loc[df["Sales"] == 300, ["Date", "Sales"]]
```

	Date	Sales
1	April-11	300
4	April-14	300
5	April-16	300

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

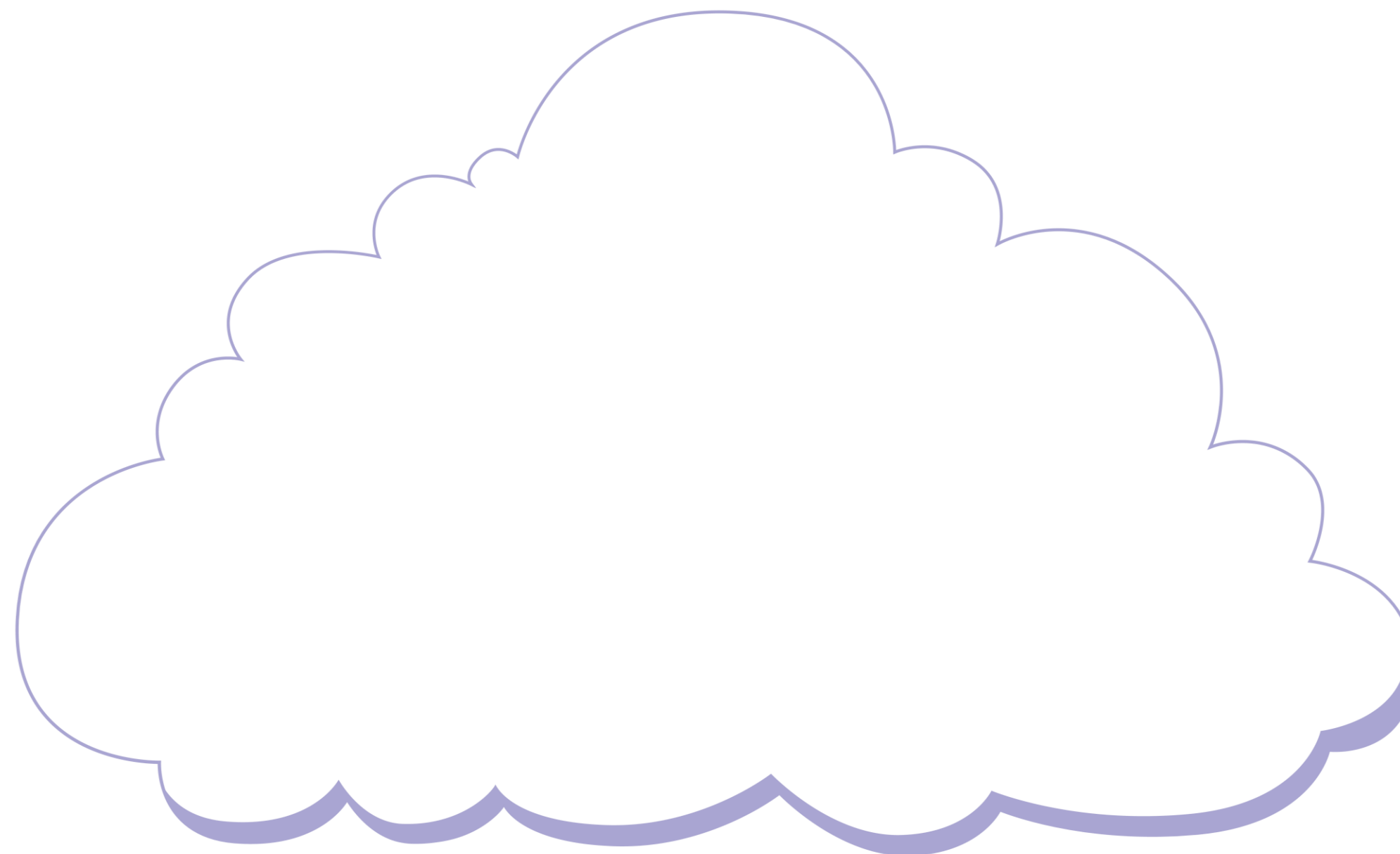
["Date", "Sales"]: Aqui, estamos especificando as colunas que queremos selecionar. Portanto, estamos filtrando as colunas "Date" e "Sales".

Portanto, a expressão `df.loc[df["Sales"] == 300, ["Date", "Sales"]]` retorna um novo DataFrame contendo apenas as linhas onde o valor da coluna "Sales" é igual a 300 e apenas as colunas "Date" e "Sales" são selecionadas.



Elaborado especialmente para o curso com apoio da ferramenta Microsoft Copilot.

Nuvem de palavras



© Getty Images

O que nós
**aprendemos
hoje?**



© Getty Images

O que nós
aprendemos
hoje?

Então ficamos assim...

- 1** Filtrar dados de um DataFrame é importante para tirar informações das tabelas.
- 2** A utilização de filtro é por linhas, colunas e por ambos ao mesmo tempo.
- 3** Pode-se usar o método `.loc` para filtrar dados de um DataFrame, selecionando linhas e/ou colunas.

Saiba mais

Quer aprender sobre como remover linhas ou colunas usando a biblioteca Pandas do Python?

Descubra em:

MATHEUS, Y. *Como remover linhas e colunas no Pandas*. Alura, 25 dez. 2019. Disponível em: <https://www.alura.com.br/artigos/como-remover-linhas-e-colunas-no-pandas>. Acesso em: 19 jul. 2024.

Referências da aula

MCKINNEY, W. *Python para análise de dados: tratamento de dados com Pandas, NumPy & Jupyter*. São Paulo: Novatec, 2023.

PANDAS. *Pandas documentation*, 10 abr. 2024. Disponível em: <https://pandas.pydata.org/docs/>. Acesso em: 19 jul. 2024.

Identidade visual: imagens © Getty Images.

Educação Profissional Paulista

Técnico em
**Ciência de
Dados**