

EXPLORANDO BIBLIOTECAS
FUNDAMENTAIS PARA CIÊNCIA
DE DADOS COM PYTHON
NUMPY • PANDAS • MATPLOTLIB

KAHEL SOUZA OLIVEIRA VICTOR LIMA VIEIRA

PROFESSORES: PROF. DR.
GIANCARLO LUCCA/ PROF. DR.
LIZANDRO OLIVEIRA





## OBJETIVO

Investigar as funcionalidades e aplicações das bibliotecas NumPy, Pandas e Matplotlib, destacando sua importância na análise e visualização de dados.

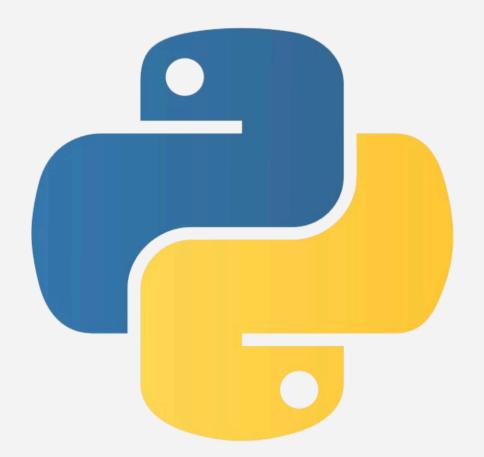






## INTRODUÇÃO

A linguagem Python tornou-se uma das principais ferramentas para ciência de dados devido à sua simplicidade e poderosas bibliotecas. Este estudo foca em três pilares: NumPy para computação numérica, Pandas para manipulação de dados e Matplotlib para visualização gráfica.







#### FERRAMENTAS UTILIZADAS

- Anaconda: distribuição que facilita a instalação de bibliotecas científicas.
- Jupyter Notebook: ambiente interativo para escrever e executar código.









## Numpy - Computação Numérica

NumPy (Numerical Python) é uma biblioteca que oferece suporte a arrays multidimensionais e operações matemáticas de alto desempenho.







#### POR QUE UTILIZAR NUMPY?

- Notimização de Código
  - NumPy é muito mais rápido que listas nativas do Python para operações numéricas.
  - Usa arrays multidimensionais otimizados em C, o que melhora a performance.
- Base para outras bibliotecas
  - pandas usa NumPy internamente para manipular dados em DataFrames.
  - matplotlib aceita arrays NumPy diretamente para gerar gráficos
- Benefícios práticos
  - Menor uso de memória
  - Operações vetoriais e matriciais com uma linha de código
  - Ideal para grandes volumes de dados





#### PYTHON PURO X NUMPY

#### Python puro:





#### PYTHON PURO X NUMPY

#### Python + Numpy:

```
Python ^
import numpy as np
# Criando uma matriz 3x3 com NumPy
matriz = np.array([[1, 2, 3],
                   [4, 5, 6],
                   [7, 8, 9]])
# Transposta da matriz
transposta = matriz.T
```

#### O que esse exemplo mostra:

- Com listas, você precisa usar duas estruturas de repetição para fazer a transposta.
- Com NumPy, basta usar .T mais limpo, mais legível, mais eficiente.
- Essa simplicidade se reflete em bibliotecas como pandas (que usa arrays NumPy para DataFrames) e matplotlib (que aceita arrays diretamente para gráficos).





#### PRINCIPAIS RECURSOS

- ndarray: estrutura de dados central.
- Operações vetorizadas.
- Funções matemáticas e estatísticas.



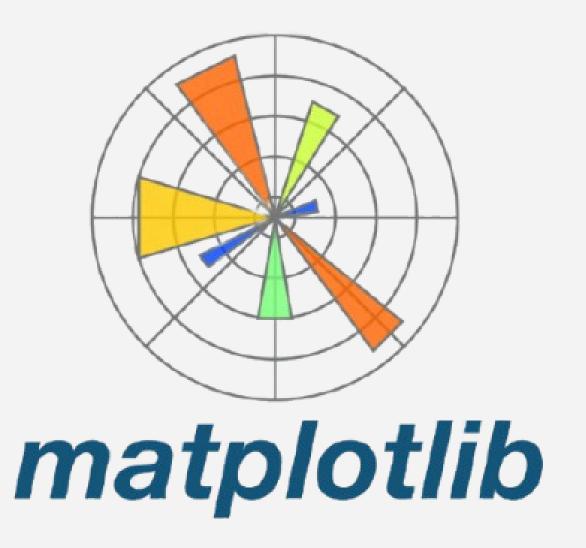


## EXEMPLO DE CÓDIGO





# MATPLOTLIB - VISUALIZAÇÃO DE DADOS Biblioteca para criar gráficos estáticos, animados e interativos.







#### PRINCIPAIS RECURSOS

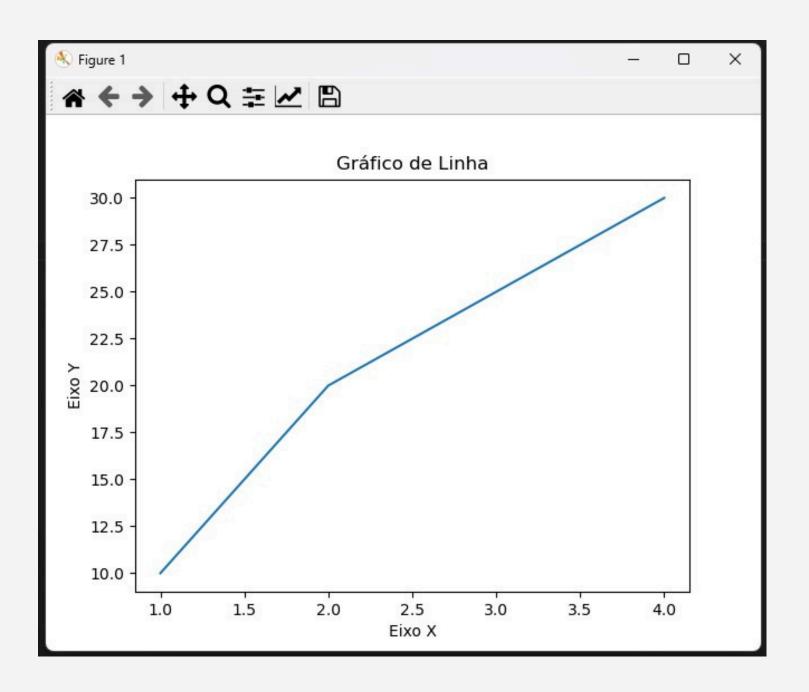
- plot(): cria um gráfico de linha.
- title(), xlabel(), ylabel(): adicionam informações ao gráfico.
- show(): exibe o gráfico.





## EXEMPLO DE CÓDIGO

```
🌳 iniciacao.py 🗙
C: > Users > Home > 💠 iniciacao.py > ...
       import matplotlib.pyplot as plt
      X = [1, 2, 3, 4]
      y = [10, 20, 25, 30]
       plt.plot(x, y)
       plt.title("Gráfico de Linha")
       plt.xlabel("Eixo X")
       plt.ylabel("Eixo Y")
       plt.show()
 10
```

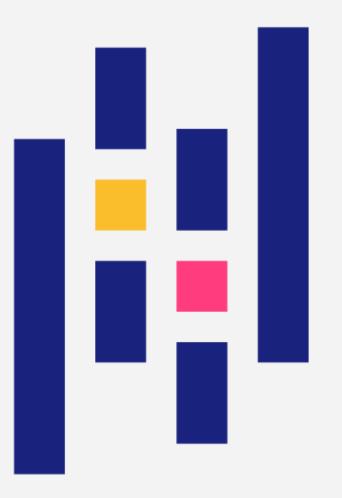






## PANDAS - MANIPULAÇÃO DE DADOS TABULARES

• Biblioteca para análise de dados em formato de tabelas (DataFrames).







### É NECESSÁRIO INSTALAR PANDAS?

Há duas formas de instalar o pandas:

- Através do site oficial: Clique aqui para instalar
- Através do terminal (exemplo abaixo: VScode)

```
Bash ^
pip install pandas
```





## QUAL DIFERENÇA DE INSTALAÇÃO?

Método	Como funciona	Quando é usado	Vantagens	Desvantagens
Via navegador	Você baixa o pacote manualmente (ex: arquivo .whl ou .zip) do site oficial como PyPI	Usado quando há problemas com o pip, ou em ambientes offline	Controle total sobre o arquivo baixado	Mais trabalhoso: precisa instalar manualmente com pip install nome_do_arquivo.whl
Via terminal (pip)	Você digita pip install pandas e o Python baixa e instala automaticamente	Método padrão e recomendado	Rápido, automático, resolve dependências sozinho	Requer conexão com a internet e o pip funcionando





#### COMO UTILIZAR PANDAS EM PYTHON?

Basicamente, para utilizar a biblioteca Pandas em Python, é necessário chamar a biblioteca, assim como na imagem abaixo: pd é um "apelido" que damos ao pandas.

```
pandas.py U X

apresentação > bibliotecas_python.py > pandas.py

1 import pandas as pd
```





#### PRINCIPAIS RECURSOS

- DataFrame: estrutura de dados em forma de tabela.
- pd.DataFrame(): cria um DataFrame.
- df[]: acessa uma coluna.
- mean(), sum(), describe(): funções estatísticas.





## EXEMPLO DE CÓDIGO

```
iniciacao.py X
C: > Users > Home > 🌵 iniciacao.py > ...
      import pandas as pd
  3 dados = {'Nome': ['Ana', 'Bruno'], 'Idade': [23, 35]}
  4 df = pd.DataFrame(dados)
      print(df)
  7 print(df['Idade'].mean()) # Média das idades
          OUTPUT DEBUG CONSOLE TERMINAL PORTS
PROBLEMS
(base) PS C:\Users\Home\AppData\Local\Programs\Microsoft VS Code> & C:\Users\Home\anaconda3\python.exe c:/Users/Home/iniciacao.py
    Nome Idade
            23
    Ana
            35
1 Bruno
29.0
(base) PS C:\Users\Home\AppData\Local\Programs\Microsoft VS Code>
```





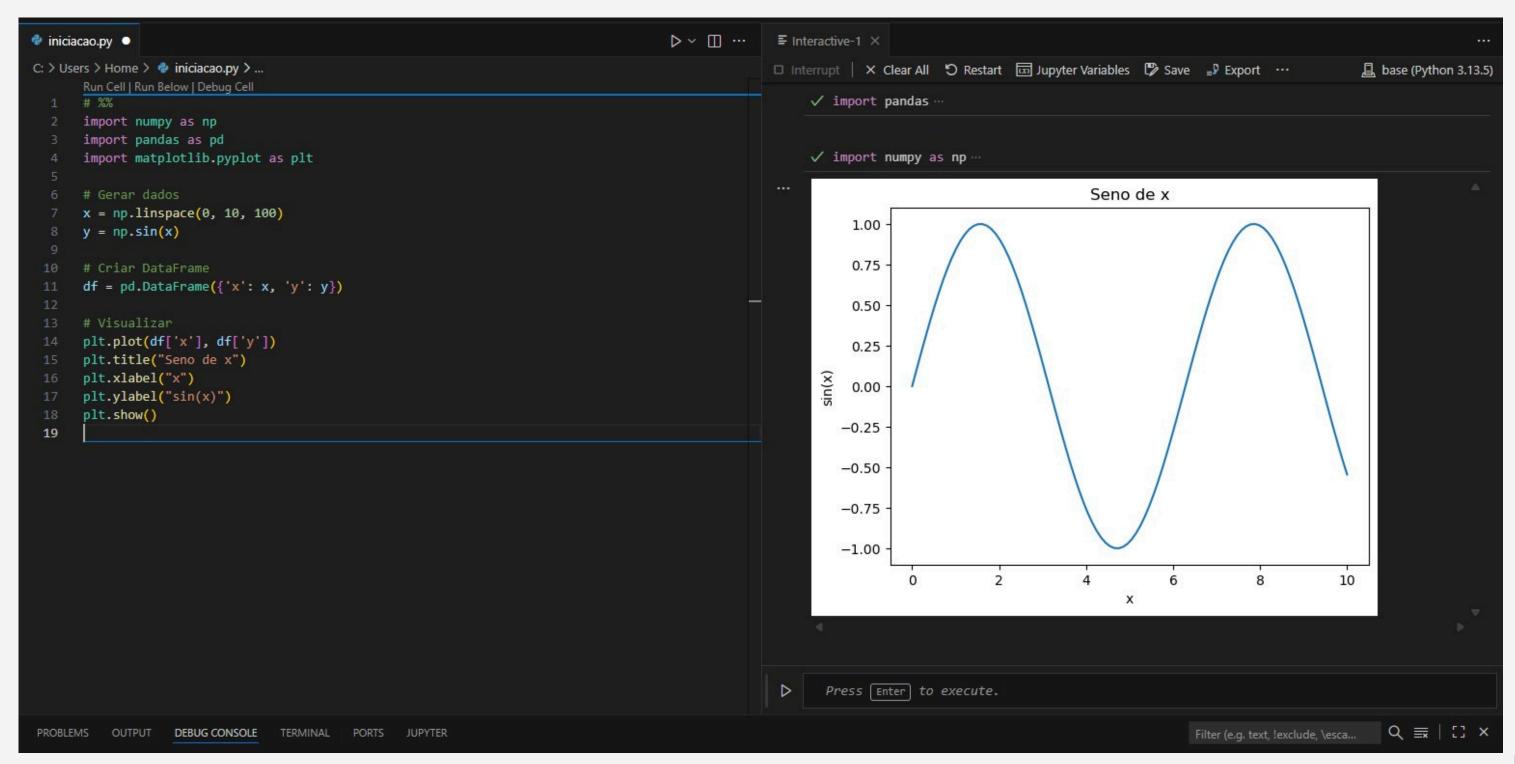
## COMPARANDO BIBLIOTECAS

BIBLIOTECA	FOCO PRINCIPAL	ESTRUTURA - CHAVE	EXEMPLO DE USO
NumPy Cálculo numérico		ndarray	Álgebra linear
Matplotlib	Visualização de dados	plot, figure	Gráficos
Pandas	Análise de dados	DataFrame	Tabelas CSV





## APLICAÇÃO INTEGRADA (JUPYTER + ANACONDA)







#### Referências

InseSTACK OVERFLOW. Instalação bibliotecas Jupyter Notebook. Disponível em: https://pt.stackoverflow.com/questions/470541/instala%c3%a7%c3%a3o-bibliotecas-jupyter-notebook...

Acesso em: 19 out. 2025.

AMARAL, Kelvyn. Introdução às bibliotecas para EDA. GitHub, 2023. Disponível em: https://github.com/KelvynAmaral/Introducao-bibliotecas-eda/blob/main/README.md.

.Acesso em: 19 out. 2025.

DIO. Python e Ciência de Dados: Explorando Pandas, NumPy e Matplotlib. Disponível em: https://www.dio.me/articles/python-e-ciencia-de-dados-explorando-pandas-numpy-e-matplotlib.

Acesso em: 19 out. 2025.

YOUTUBE. Curso de Python para análise de dados. Disponível em: https://www.youtube.com/watch?

v=9Cw7iljFlBc&list=PLvlkVRRKOYFQHnDhjTmXLEz3HU5WTgOcF.

.Acesso em: 19 out. 2025.

PANDAS. Pandas: Python Data Analysis Library. Disponível em: https://pandas.pydata.org/.

.Acesso em: 19 out. 2025.

NUMPY. NumPy: The fundamental package for scientific computing with Python. Disponível em: https://numpy.org/...

Acesso em: 19 out. 2025.

MATPLOTLIB. Matplotlib: Visualization with Python. Disponível em: https://matplotlib.org/.

. Acesso em: 19 out. 2025.













f (m) /ucpel



Juntos por novas possibilidades