

# Serpente Arcana

Código e Destino em Python



Domine os segredos dos dados e desperte o poder do  
Python na Ciência de Dados

Lídia Bahia

# INTRODUÇÃO

Python se tornou uma das linguagens mais importantes no ecossistema de **Ciência de Dados**. Ela combina simplicidade na sintaxe com um conjunto robusto de bibliotecas que permitem:

- ✓ manipular dados,
- ✓ criar visualizações,
- ✓ treinar modelos de Machine Learning,
- ✓ e integrar análises usando SQL.

Este material apresenta uma visão objetiva das principais ferramentas no fluxo de trabalho em Data Science com Python.

# 1. Por que Python na Ciência de Dados?

Python é adotado no mundo inteiro por analistas, cientistas de dados e engenheiros por três motivos principais:

- ✓ Curva de aprendizado amigável
- ✓ Comunidade enorme e ativa
- ✓ Ecossistema rico de pacotes orientados a dados

Algumas áreas em que Python se destaca:

- ✓ Análise exploratória de dados (EDA),
- ✓ Automação de processos,
- ✓ Engenharia de atributos,
- ✓ Modelos de previsão,
- ✓ Dashboards e storytelling de dados

## 2. Pacotes Essenciais no Fluxo de Análise de Dados

✓ Manipulação e Análise de Dados:

Biblioteca	Uso Principal
Pandas	Leitura, limpeza, transformação e análise de dados tabulares
Numpy	Operações matemáticas e vetorização de dados
Polars (alternativa opcional)	Consultas mais rápidas em datasets grandes

## ✓ Visualização e Storytelling de Dados:

Biblioteca	O que oferece
Matplotlib	Gráficos base, personalizáveis
Seaborn	Visualizações estatísticas de alto nível
Plotly	Gráficos interativos e dashboards
Bokeh / Altair (opcionais)	Visualizações orientadas a insights rápidos

Exemplo simples de código para geração de gráfico:

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.scatterplot(data=df, x="idade", y="renda")
plt.title("Relação entre idade e renda")
plt.show()
```

# 3. Uso de SQL dentro do Python

A integração entre Python e SQL é fundamental para cessar e analisar dados armazenados em bancos.

✓ Formas comuns de usar SQL no Python:

Abordagem	Como funciona
SQLite + SQLAlchemy	Banco local embutido em arquivo .db
Pandas + read_sql()	Executa queries direto no DataFrame
DuckDB	Permite consultas SQL em arquivos CSV e Parquet com alta performance
Modulo sql do Jupyter	Permite escrever queries com %%sql em notebooks

## ✓ Exemplo: SQL com Pandas:

```
import pandas as pd
import sqlite3

conn = sqlite3.connect("vendas.db")

query = """
SELECT produto, SUM(valor) AS total_vendas
FROM vendas
GROUP BY produto
ORDER BY total_vendas DESC
"""

df_vendas = pd.read_sql(query, conn)
df_vendas.head()
```

# 4. Pipeline simples de Data Science com Python

## Etapas comuns do projeto

### 1. Coleta e conexão com fontes de dados

CSV, SQL, APIs, web scraping...

### 2. Limpeza e transformação

Tratamento de valores ausentes, padronizações, conversões

### 3. Análise exploratória (EDA)

Visualizações, correlações, estatísticas descritivas

### 4. Modelagem (opcional)

Modelos preditivos com scikit-learn

### 5. Apresentação e entrega

Relatórios, dashboards, notebooks reprodutíveis

# 4. Boas práticas

- ✓ Usar ambientes virtuais (venv/conda)
- ✓ Organização de pastas e scripts
- ✓ Documentação mínima do código
- ✓ Versionamento com Git e GitHub

## CONCLUSÃO

Python é uma ferramenta completa para Ciência de Dados, permitindo ir do SQL até a modelagem e visualização de forma integrada.

Dominar os pacotes certos e entender o fluxo de análise de dados torna seu trabalho mais eficiente e suas entregas mais profissionais.