

# O IMPÉRIO DOS DADOS EM PYTHON



Maurício Nazario

# O Império dos Dados em Python

Neste eBook, você vai aprender de forma simples e direta como usar Python para analisar, interpretar e visualizar dados. A ideia é que cada capítulo traga exemplos reais e aplicáveis, que podem ser usados tanto em estudos quanto em projetos profissionais.

## 1. O Poder dos Dados

Vivemos em um mundo movido por dados. Empresas, governos e até pequenos negócios tomam decisões baseadas em informações extraídas de números, planilhas e sistemas. Python é uma das principais ferramentas para isso.

**Exemplo prático:** calcular a média de consumo de energia de clientes.

```
import pandas as pd
dados = pd.DataFrame({ 'Cliente': ['A', 'B', 'C', 'D'], 'Consumo_kWh': [320, 250, 410, 290] })
media = dados['Consumo_kWh'].mean()
print(f"Média de consumo: {media:.2f} kWh")
```

## 2. Explorando Dados com Pandas

A biblioteca **Pandas** permite manipular e explorar grandes volumes de dados de forma rápida e eficiente. É como ter o poder do Excel, mas com muito mais controle.

**Exemplo prático:** encontrar o produto mais vendido de uma loja.

```
import pandas as pd
vendas = pd.DataFrame({ 'Produto':
['Mouse', 'Teclado', 'Monitor', 'Mouse', 'Monitor'],
'Quantidade': [10, 8, 5, 7, 6] })
mais_vendido = vendas.groupby('Produto')['Quantidade'].sum().sort_values(ascending=False)
print(mais_vendido.head(1))
```

### 3. Visualizando Dados com Matplotlib

Ver dados é uma das formas mais eficazes de entender o que está acontecendo.

O **Matplotlib** nos ajuda a criar gráficos e revelar padrões de forma visual.

**Exemplo prático:** gráfico simples de faturamento mensal.

```
import matplotlib.pyplot as plt meses = ['Jan', 'Fev', 'Mar',  
'Abr'] faturamento = [12000, 15000, 17000, 16000]  
plt.plot(meses, faturamento, marker='o', color='blue')  
plt.title('Faturamento Mensal') plt.xlabel('Mês')  
plt.ylabel('Valor (R$)') plt.grid(True) plt.show()
```

## 4. Tratando Dados Ausentes

Em qualquer base de dados real, é comum encontrar valores ausentes ou incorretos. Tratar esses dados é essencial para garantir análises precisas.

**Exemplo prático:** substituindo valores ausentes por médias.

```
import pandas as pd
dados = pd.DataFrame({ 'Aluno': ['Ana', 'Bruno', 'Carla', 'Daniel'], 'Nota': [8.5, None, 7.0, None] })
dados['Nota'].fillna(dados['Nota'].mean(), inplace=True)
print(dados)
```

## 5. Conclusão: O Império é Seu

Agora você já tem as bases para construir suas próprias análises e dominar o império dos dados. Python oferece um universo de possibilidades — e a melhor forma de aprender é praticando!

Continue explorando bibliotecas como **NumPy**, **Seaborn** e **Scikit-Learn** para elevar suas habilidades em análise e ciência de dados ao próximo nível.