**Cartilha Prática de Matplotlib**

**✨ Introdução**

Matplotlib é uma biblioteca de visualização de dados em Python amplamente usada para criar gráficos 2D (e alguns 3D). A interface mais comum é o módulo pyplot, importado como plt.

**🔍 Importação da biblioteca**

import matplotlib.pyplot as plt

**📊 Configurações Básicas**

**Criar figura:**

plt.figure(figsize=(largura, altura))

**Título e rótulos:**

plt.title("Título do Gráfico")

plt.xlabel("Nome do Eixo X")

plt.ylabel("Nome do Eixo Y")

**Mostrar gráfico:**

plt.show()

**Salvar gráfico:**

plt.savefig("nome\_do\_arquivo.png", dpi=300, bbox\_inches='tight')

**Estilos prontos:**

plt.style.use('ggplot') # Outros: 'seaborn', 'bmh', 'fivethirtyeight', etc.

**📈 Gráfico de Linha**

x = [1, 2, 3, 4, 5]

y = [10, 20, 15, 25, 30]

plt.plot(x, y, color='blue', linestyle='--', marker='o', linewidth=2)

plt.title("Gráfico de Linha")

plt.xlabel("Eixo X")

plt.ylabel("Eixo Y")

plt.grid(True)

plt.show()

**📅 Gráfico de Barras**

categorias = ['A', 'B', 'C', 'D']

valores = [10, 15, 7, 12]

plt.bar(categorias, valores, color='green', edgecolor='black')

plt.title("Gráfico de Barras")

plt.xlabel("Categorias")

plt.ylabel("Valores")

# Adiciona valores acima das barras

for i, valor in enumerate(valores):

plt.text(i, valor + 0.5, str(valor), ha='center')

plt.show()

**🍕 Gráfico de Pizza**

labels = ['Produto A', 'Produto B', 'Produto C']

valores = [40, 35, 25]

plt.pie(valores, labels=labels, autopct='%1.1f%%', startangle=90)

plt.title("Distribuição de Produtos")

plt.axis('equal') # Deixa o círculo redondo

plt.show()

**🔍 Gráfico de Dispersão (Scatter Plot)**

x = [5, 7, 8, 7, 2, 17, 2, 9, 4, 11]

y = [99, 86, 87, 88, 100, 86, 103, 87, 94, 78]

plt.scatter(x, y, color='red', marker='x')

plt.title("Gráfico de Dispersão")

plt.xlabel("Variável X")

plt.ylabel("Variável Y")

plt.grid(True)

plt.show()

**🌐 Subplots (Múltiplos gráficos em uma figura)**

fig, axs = plt.subplots(1, 2, figsize=(10, 4))

axs[0].bar(['A', 'B', 'C'], [5, 7, 6])

axs[0].set\_title("Gráfico 1")

axs[1].plot([1, 2, 3], [4, 5, 6])

axs[1].set\_title("Gráfico 2")

plt.tight\_layout()

plt.show()

**✨ Anotações e Destaques**

**Texto:**

plt.text(x=2, y=25, s="Ponto de destaque", fontsize=12)

**Anotação com seta:**

plt.annotate('Pico', xy=(3, 30), xytext=(2, 35),

arrowprops=dict(facecolor='black', arrowstyle='->'))

**🔗 Dicas Extras**

* Use plt.grid(True) para melhorar a leitura dos gráficos
* plt.legend() para adicionar legendas quando houver mais de uma linha
* Use plt.xticks() e plt.yticks() para personalizar os eixos
* Gráficos podem ser combinados com twinx() para dois eixos Y

**📄 Conclusão**

Com matplotlib, é possível criar gráficos informativos, profissionais e personalizados para relatórios, dashboards ou exploração de dados. Continue praticando e combinando os gráficos com outras bibliotecas como pandas e numpy!

Se quiser, posso transformar essa cartilha em PDF, slide ou incluir mais exemplos avançados. ✌️