

Módulo 6 - Visualização de dados 1/15

6.1 - O que é uma boa visualização de dados
VIDEO - 8 MIN6.2 - Tipos de gráficos e elementos visuais
VIDEO - 10 MINExercício de Revisão | Gráficos
QUIZ - 1 PERGUNTA6.3 - Visualização de dados em python (parte 1)
VIDEO - 16 MIN6.3 - Visualização de dados em python (parte 2)
VIDEO - 20 MINExercício de Código | Visualização de dados em python
TEXTFaça você mesmo | Visualização de dados em python
TEXT6.4 - Ferramentas de visualização de dados
VIDEO - 4 MIN

6.5 - Elaborando o rascunho do nosso

Exercício de Código | Visualização de dados em python

4 COMENTÁRIOS/DÚVIDAS

Exercício de Código

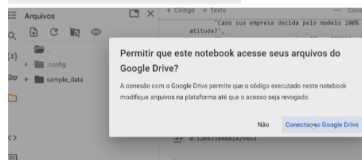
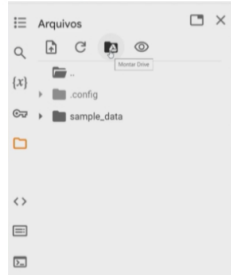
Vamos fazer nossos gráficos e a visualização de dados, assim como nossa instrutora realizou na última aula 🤗

Acompanhe o gabarito:

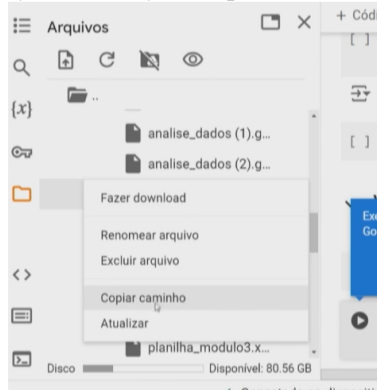
Parte 1:

Vamos fazer a leitura da nossa tabela.

Conecte o Google Drive:



Copie o caminho do seu arquivo de análise_dados:



Importe a biblioteca e faça a leitura colando o caminho do arquivo:

```
[ ] 1 from google.colab import drive
    2 drive.mount('/content/drive')

[1] 1 import pandas as pd

    1 = pd.read_csv('/content/drive/MyDrive/programaria/ana
```

Vamos criar um gráfico de barras para visualizar a diferença entre os gêneros:

```
[7] 1 genero_counts = dados['GENERO'].value_counts()

[6] 1 import matplotlib.pyplot as plt

    1 plt.figure()
    2 plt.bar(height = genero_counts.values, x = genero_counts.index)
    3 plt.title('Quantidade de pessoas por gênero na area de Dados')
    4 plt.xlabel('Gênero')
    5 plt.ylabel('Contagem')
    6 plt.show()
```

Outra forma de fazer esse mesmo gráfico com outra biblioteca:

```
[24] 1 import seaborn as sns

    1 plt.figure()
    2 sns.countplot(data = dados, x = 'GENERO', palette = 'pastel')
    3 plt.title('Quantidade de pessoas por gênero na area de Dados')
    4 plt.xlabel('Gênero')
    5 plt.ylabel('Contagem')
    6 plt.grid(True)
    7 plt.show()
```

Parte 2:

Vamos verificar se a média do salário varia de acordo com a idade e criar um gráfico:

```
[27] 1 salario_por_idade = dados.groupby("IDADE")["SALARIO"].mean()

1 plt.figure()
2 plt.plot(salario_por_idade.index, salario_por_idade.values, marker='o', linestyle='--')
3 plt.xlabel('Idade')
4 plt.ylabel('Média de salário')
5 plt.title('Média de salário por idade')
6 plt.grid(True)
7 plt.show()
```

Vamos usar `plotly` para criar o gráfico em outro estilo:

```
import plotly.express as px

fig = px.line(salario_por_idade.reset_index(), x='Idade', y='Salário', title='Média de Salário por Idade', markers=True)
```

Adicione `fig.show()` na linha abaixo:

```
1 fig = px.line(salario_por_idade.reset_index(), x='IDADE', y='SALARIO')
2 fig.show()
```

Vamos fazer um gráfico do tipo `scatterplot`:

```
1 plt.figure(figsize=(15,5))
2 plt.scatter(dados['IDADE'], dados['SALARIO'], alpha=0.5)
3 plt.xlabel('Idade')
4 plt.ylabel('Salário')
5 plt.title('Dispersão de salário por idade')
6 plt.grid(True)
7 plt.show()
```

Vamos usar `scatter` para criar o gráfico em outro estilo:

```
1 fig = px.scatter(dados, x='IDADE', y='SALARIO', title='Dispersão de salário por idade')
2 fig.show()
```