1. Preparação do Ambiente e Estilo dos Gráficos

* **plt.style.use('seaborn-darkgrid')**: Esse comando define o estilo visual dos gráficos criados com o **matplotlib**. O estilo 'seaborn-darkgrid' aplica um fundo escuro com uma grade (grid) clara. É uma forma de deixar os gráficos mais bonitos e legíveis.
* **sns.set\_palette('deep')**: Define a paleta de cores do gráfico. O **Seaborn** oferece diferentes paletas de cores, e a 'deep' é uma paleta com cores mais intensas, ideal para gráficos com vários elementos.

1. Exploração e Limpeza dos Dados

* **df.isnull().sum()**: Esse comando verifica se existem valores ausentes (nulos) no DataFrame. **isnull()** retorna um DataFrame de valores booleanos (True ou False) indicando se o valor é nulo. **sum()** conta quantos valores nulos existem em cada coluna.
* **df.duplicated().sum()**: Esse comando verifica se existem linhas duplicadas no DataFrame. O **duplicated()** retorna um vetor de booleanos (True ou False) indicando se a linha é uma duplicata. **sum()** conta quantas duplicatas existem.
* **df.info()**: Esse comando retorna informações gerais sobre o DataFrame, como o número de entradas, as colunas presentes, o tipo de dados de cada coluna e quantos valores não nulos existem.
* **df.describe()**: Esse comando gera estatísticas descritivas para as colunas numéricas do DataFrame, como média, desvio padrão, valores mínimos e máximos, etc.

1. Limpeza de Texto

* **stopwords.words('portuguese')**: O **NLTK** (Natural Language Toolkit) fornece uma lista de palavras comuns (como "e", "a", "de") que são chamadas de *stopwords*. Essas palavras não têm um valor significativo na análise, então elas foram removidas do texto para focar nas palavras mais relevantes.
* **''.join([char for char in texto if char.isalnum() or char.isspace()])**: Essa parte do código remove todos os caracteres que não são alfanuméricos (letras e números) ou espaços em branco, eliminando a pontuação.
* **texto.lower()**: Converte todo o texto para minúsculas para garantir que palavras como "Olá" e "olá" sejam tratadas da mesma forma.
* **' '.join([palavra for palavra in texto.split() if palavra not in stop\_words])**: Essa linha divide o texto em palavras, e então filtra as palavras que estão na lista de *stopwords*. O resultado é uma string de palavras relevantes, sem as stopwords.

1. Análise de Frequência de Palavras e Visualizações

* **WordCloud().generate()**: Esse comando cria uma **nuvem de palavras** onde as palavras mais frequentes são exibidas em maior tamanho. O **' '.join(df['Comentário Limpo'])** combina todos os textos da coluna 'Comentário Limpo' em uma única string, que é passada para o **WordCloud** gerar a nuvem.
* **plt.imshow(wordcloud, interpolation='bilinear')**: Exibe a nuvem de palavras gerada, com interpolação bilinear para suavizar a imagem.
* **plt.axis('off')**: Remove os eixos do gráfico, pois não são necessários para a visualização da nuvem de palavras.
* **' '.join(df['Comentário Limpo']).split()**: Junta todos os comentários limpos em uma única string e depois divide em uma lista de palavras.
* **Counter(palavras)**: Conta a frequência de cada palavra na lista e armazena no formato de dicionário, onde a chave é a palavra e o valor é a contagem.

1. Análise de Sentimentos

* **TextBlob(texto).sentiment.polarity**: O **TextBlob** é uma ferramenta de análise de sentimentos que retorna a *polaridade* do texto, um valor entre -1 (muito negativo) e 1 (muito positivo). Valores próximos de 0 indicam sentimentos neutros.
* **df['Comentário Limpo'].apply(analisar\_sentimento)**: Aplica a função analisar\_sentimento() em cada comentário da coluna 'Comentário Limpo', gerando uma nova coluna 'Sentimento' com a polaridade dos sentimentos.
* **lambda x: 'Positivo' if x > 0 else ('Negativo' if x < 0 else 'Neutro')**: Esse lambda classifica a polaridade do sentimento: se for maior que 0, é positivo; se menor que 0, é negativo; e se for igual a 0, é neutro.
* **sns.countplot(x='Classificação Sentimento', data=df)**: Cria um gráfico de barras mostrando a contagem dos diferentes sentimentos (Positivo, Negativo, Neutro).