

▼ Análise Exploratória. Jailson

```
#Importando as bibliotecas
import pandas as pd
import matplotlib.pyplot as plt
plt.style.use("seaborn")
```

```
#Upload do arquivo
from google.colab import files
arq = files.upload()
```

Escolher arquivos

AdventureWorks.xlsx

- **AdventureWorks.xlsx**(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 85328 bytes, last modified: 09/06/2020 - 100% done  
Saving AdventureWorks.xlsx to AdventureWorks (1).xlsx

```
#Criando nosso DataFrame
df = pd.read_excel("/AdventureWorks.xlsx")
```

```
#Visualiza as 5 primeiras linhas
df.head(5)
```

	Data Venda	Data Envio	ID Loja	ID Produto	ID Cliente	No. Venda	Custo Unitário	Preço Unitário	Quantidade
0	2008-05-09	2008-05-29	199	384	18839	200805093CS607	348.58	758.0	
1	2008-05-12	2008-05-17	306	384	19051	200805123CS567	348.58	758.0	
2	2008-05-14	2008-05-20	306	384	19052	200805143CS576	348.58	758.0	
3	2008-05-21	2008-05-27	306	384	19052	200805213CS576	348.58	758.0	
4	2008-06-20	2008-06-27	306	384	19053	200806203CS586	348.58	758.0	

```
#Quatidade de Linhas
df.shape
```

```
ut.snape
```

```
(904, 16)
```

```
#Tipos de dados
```

```
df.dtypes
```

```
Data Venda      datetime64[ns]
Data Envio       datetime64[ns]
ID Loja          int64
ID Produto       int64
ID Cliente       int64
No. Venda        object
Custo Unitário   float64
Preço Unitário   float64
Quantidade       int64
Valor Desconto   float64
Valor Venda      float64
Produto          object
Fabricante       object
Marca            object
Classe           object
Cor              object
dtype: object
```

```
#Qual a Receita total?
```

```
df["Valor Venda"].sum()
```

```
5984606.1426
```

```
#Custo Total
```

```
df["custo"] = df["Custo Unitário"].mul(df["Quantidade"]) #Criando a coluna
```

```
df.head(1)
```

ID Cliente	No. Venda	Custo Unitário	Preço Unitário	Quantidade	Valor Desconto	Valor Venda	Prod
18839	200805093CS607	348.58	758.0	6	0.0	4548.0	Advent Wc Laptop15. M1548 F

```
#Qual o Custo total?
```

```
round(df["custo"].sum(), 2)
```

```
2486783.05
```

```
#Agora que temos a receita e custo e o total, podemos achar o Lucro total
```

```
#Vamos criar uma coluna de Lucro que será Receita - Custo
```

```
df["lucro"] = df["Valor Venda"] - df["custo"]
```

```
df.head(1)
```

	No. Venda	Custo Unitário	Preço Unitário	Quantidade	Valor Desconto	Valor Venda	Produto	Fabr
200805093CS607		348.58	758.0	6	0.0	4548.0	Adventure Works Laptop15.4W M1548 Red	Ad

```
#Total Lucro
round(df["lucro"].sum(),2)
```

```
3497823.09
```

```
#Criar coluna com total de dias para enviar o produto
df["Tempo_envio"] = df["Data Envio"] - df["Data Venda"]
```

```
df.head(1)
```

	Custo Unitário	Preço Unitário	Quantidade	Valor Desconto	Valor Venda	Produto	Fabricante	Ma
S607	348.58	758.0	6	0.0	4548.0	Adventure Works Laptop15.4W M1548 Red	Adventure Works	Adven Wi

▼ \*\*Agora, queremos saber a média do tempo de envio para cada marca, e para isso precisamos transformar a coluna Tempo\_envio em numérica

```
#Extrindo apenas os dias
df["Tempo_envio"] = (df["Data Envio"] - df["Data Venda"]).dt.days
```

```
df.head(1)
```

```

      Data  Data  ID  ID  ID  No. Venda  Custo  Preço  Quantid
#Verificar o tipo da coluna Tempo_envio
df["Tempo_envio"].dtype

dtype('int64')

```

```

#Média do tempo de envio por marca
df.groupby("Marca")["Tempo_envio"].mean()

```

```

Marca
Adventure Works    8.663866
Contoso            8.470930
Fabrikam           8.510121
Name: Tempo_envio, dtype: float64

```

```

#Verificar se temos dados faltantes
df.isnull().sum()

```

```

Data Venda      0
Data Envio      0
ID Loja         0
ID Produto      0
ID Cliente      0
No. Venda       0
Custo Unitário  0
Preço Unitário  0
Quantidade      0
Valor Desconto  0
Valor Venda     0
Produto         0
Fabricante      0
Marca           0
Classe          0
Cor             0
custo           0
lucro           0
Tempo_envio     0
dtype: int64

```

## ▼ \*Saber o Lucro por Ano e Por Marca?

```

#Vamos Agrupar por ano e marca
df.groupby([df["Data Venda"].dt.year, "Marca"])[ "lucro"].sum()

```

```

Data Venda  Marca
2008        Adventure Works    306,641.16
           Contoso             56,416.00
           Fabrikam            1,557,020.55
2009        Adventure Works    405,395.08
           Contoso             138,258.95
           Fabrikam            1,034,091.35
Name: lucro, dtype: float64

```

```
pd.options.display.float_format = '{:20,.2f}'.format #Melhora as numerações cient
```

```
#Resetando o index
```

```
lucro_ano = df.groupby([df["Data Venda"].dt.year, "Marca"])[ "lucro"].sum().reset_
lucro_ano
```

	Data Venda	Marca	lucro
0	2008	Adventure Works	306,641.16
1	2008	Contoso	56,416.00
2	2008	Fabrikam	1,557,020.55
3	2009	Adventure Works	405,395.08
4	2009	Contoso	138,258.95
5	2009	Fabrikam	1,034,091.35

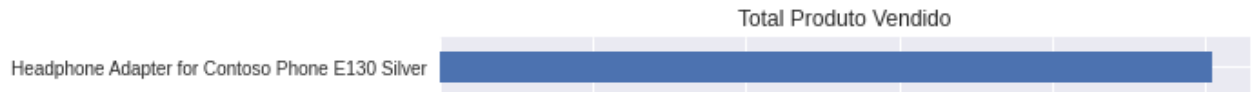
```
#Qual o total de produto vendido
```

```
df.groupby("Produto")["Quantidade"].sum().sort_values(ascending=False)
```

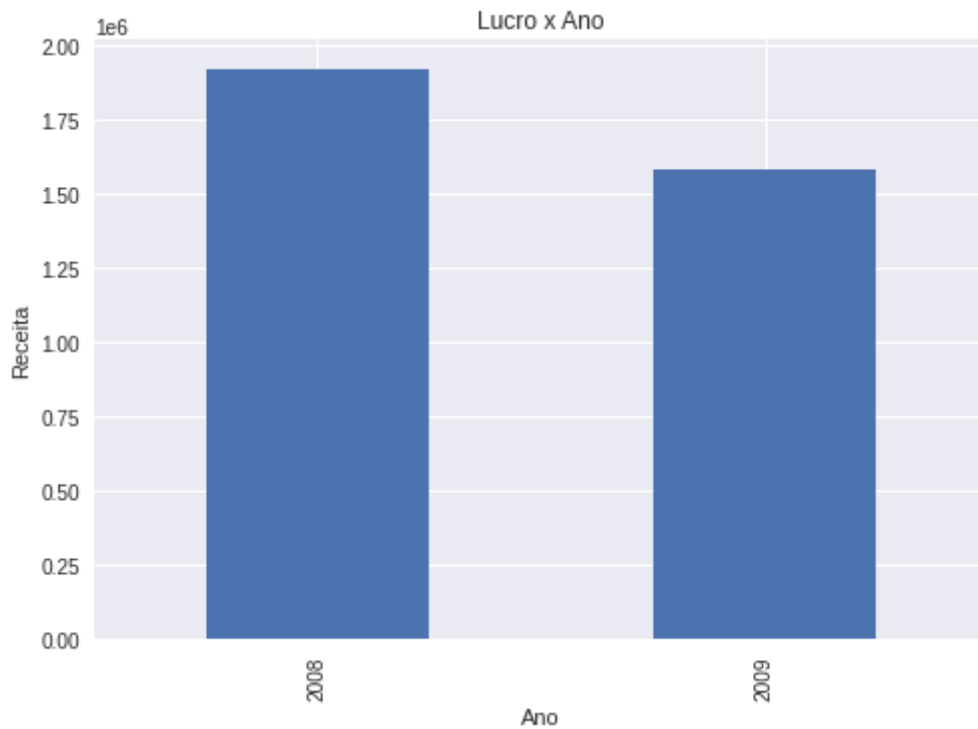
```
Produto
Headphone Adapter for Contoso Phone E130 Silver    25232
Headphone Adapter for Contoso Phone E130 White    25008
Adventure Works Laptop15.4W M1548 Black           1089
Fabrikam Trendsetter 2/3'' 17mm X100 Grey          1087
Adventure Works Laptop15.4W M1548 Red             1047
Fabrikam Trendsetter 2/3'' 17mm X100 Black          926
Fabrikam Trendsetter 1/3'' 8.5mm X200 Black         884
Fabrikam Trendsetter 1/3'' 8.5mm X200 Grey          845
Fabrikam Trendsetter 1/3'' 8.5mm X200 White         789
Name: Quantidade, dtype: int64
```

```
#Grafico Total de produto vendido
```

```
df.groupby("Produto")["Quantidade"].sum().sort_values(ascending=True).plot.barh(t
plt.xlabel("Total")
plt.ylabel("Produto");
```



```
df.groupby(df["Data Venda"].dt.year)["lucro"].sum().plot.bar(title="Lucro x Ano")
plt.xlabel("Ano")
plt.ylabel("Receita");
```



```
df.groupby(df["Data Venda"].dt.year)["lucro"].sum()
```

```
Data Venda
2008      1,920,077.71
2009      1,577,745.38
Name: lucro, dtype: float64
```

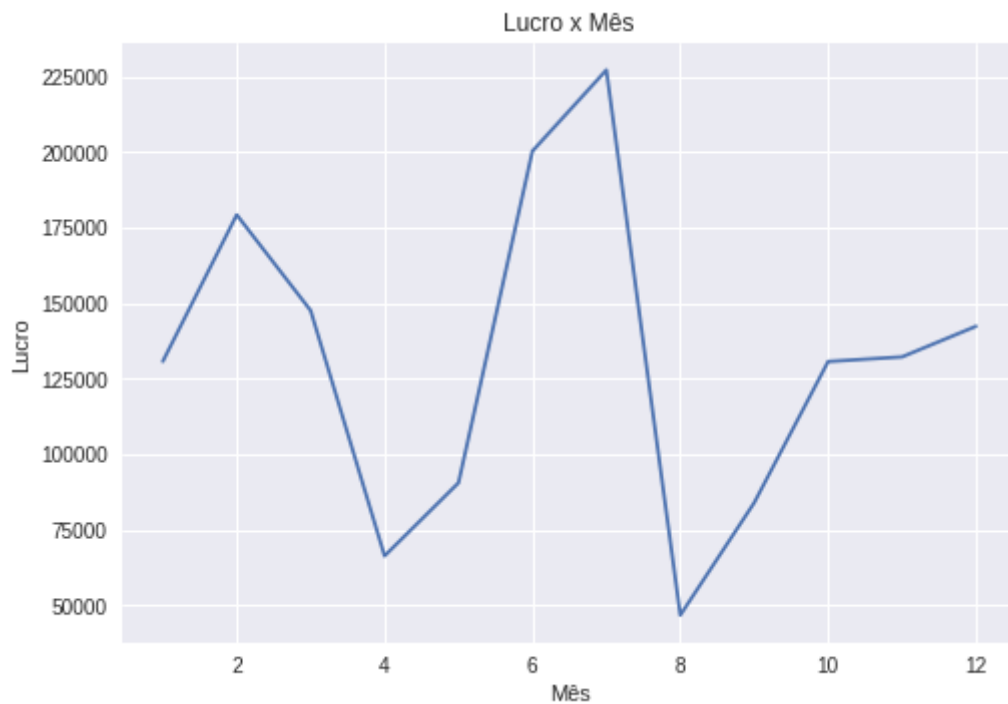
```
#Selecionando apenas as vendas de 2009
df_2009 = df[df["Data Venda"].dt.year == 2009]
```

```
df_2009.head()
```

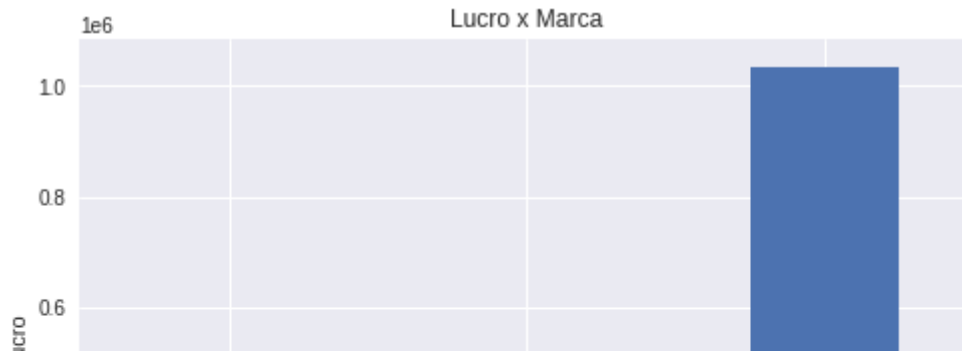
	Data Venda	Data Envio	ID Loja	ID Produto	ID Cliente	No. Venda	Custo Unitário	Preço Unitário	Quantidade
11	2009-05-02	2009-05-14	199	384	18938	200905023CS847	348.58	758.00	
12	2009-05-16	2009-05-27	306	384	19067	200905163CS746	348.58	758.00	
13	2009-05-18	2009-05-29	306	384	19068	200905183CS748	348.58	758.00	

#Saber o meu lucro por mes

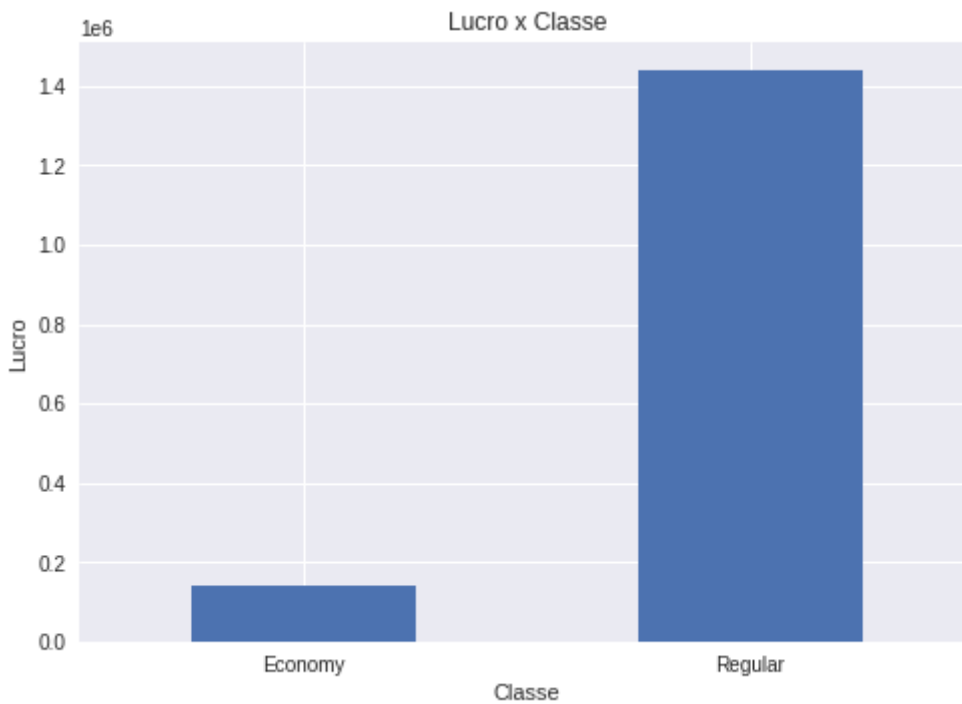
```
df_2009.groupby(df_2009["Data Venda"].dt.month)["lucro"].sum().plot(title="Lucro
plt.xlabel("Mês")
plt.ylabel("Lucro");
```



```
df_2009.groupby("Marca")["lucro"].sum().plot.bar(title="Lucro x Marca")
plt.xlabel("Mês")
plt.ylabel("Lucro")
plt.xticks(rotation='horizontal');
```



```
#lucro por classe
df_2009.groupby("Classe")["lucro"].sum().plot.bar(title="Lucro x Classe")
plt.xlabel("Classe")
plt.ylabel("Lucro");
plt.xticks(rotation='horizontal');
```



```
df["Tempo_envio"].describe() #analise estatisticas
```

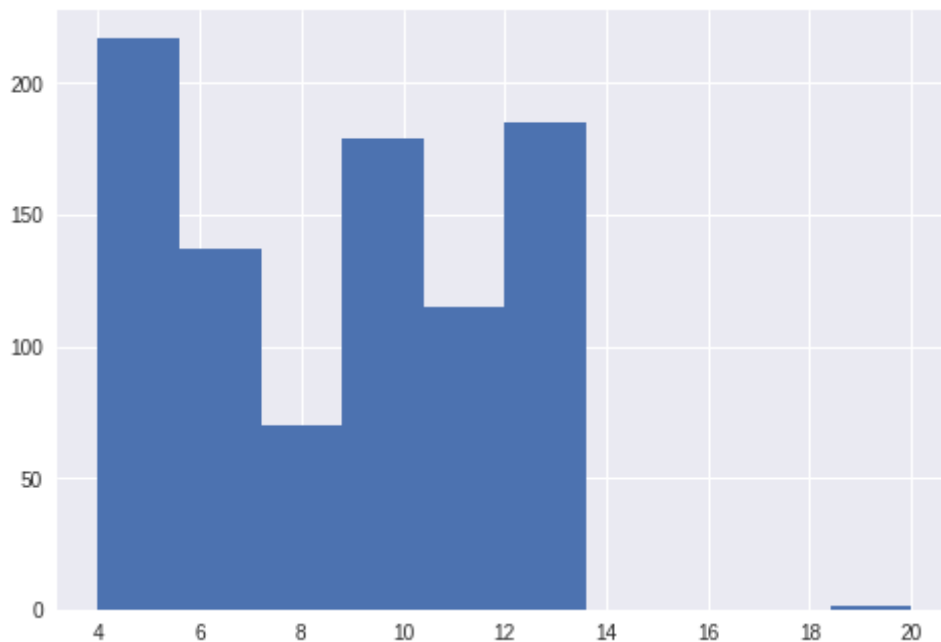
```
count          904.00
mean            8.54
std             3.06
min             4.00
25%             6.00
50%             9.00
75%            11.00
max            20.00
Name: Tempo_envio, dtype: float64
```

```
#Grafico de Boxplot
plt.boxplot(df["Tempo_envio"]);
```





```
#Histograma
plt.hist(df["Tempo_envio"]);
```



```
#Tempo minimo de envio
df["Tempo_envio"].max()
```

```
20
```

```
df[df["Tempo_envio"] ==20] #Identificando o Outlier
```

	Data Venda	Data Envio	ID Loja	ID Produto	ID Cliente	No. Venda	Custo Unitário	Preço Unitário	Quantic
0	2008-05-09	2008-05-29	199	384	18839	200805093CS607	348.58	758.00	

```
#Salvando em csv
```

```
#salvando em .csv
```

```
df.to_csv("df_vendas_novo.csv", index=False)
```

---

✓ 0s conclusão: 13:41

