**Análise de dados com Python e Pandas**

**Introdução**

Neste Laboratório da plataforma [DIO](https://www.dio.me/), presente no Bootcamp Geração Tech Unimed-BH - Ciência de Dados, é apresentado a biblioteca Pandas, uma biblioteca Python de código aberto para análise de dados. Ela dá ao Python a capacidade de trabalhar com dados do tipo planilha, permitindo carregar, manipular e combinar dados rapidamente, entre outras funções.

**Estrutura de dados**

**Listas**

animais **=** [1,2,3] *#Criando uma lista chamada animais*

animais

*[1, 2, 3]*

*animais* ***=*** *["cachorro", "gato", 12345, 6.5]*

*animais*

['cachorro', 'gato', 12345, 6.5]

animais[0] *#Imprimindo o primeiro elemento da lista*

'cachorro'

animais[3] *#Imprimindo o 4 elemento da lista*

6.5

animais[0] **=** "papagaio" *#Substituindo o primeiro elemento da lista*

animais

['papagaio', 'gato', 12345, 6.5]

animais**.**remove("gato") *#Removendo "gato" da lista*

animais

['papagaio', 12345, 6.5]

len(animais) *#Verificando o tamanho da lista*

3

"gato" **in** animais *#Verificando a existência de "gato" na lista*

False

lista **=** [500, 30, 300, 80, 10]

max(lista) *#Maior valor da lista*

500

min(lista) *#Menor valor da lista*

10

animais**.**append(["leão", "Cachorro"]) *#Cria uma lista aninhada (lista dentro de lista)*

animais

['papagaio', 12345, 6.5, ['leão', 'Cachorro']]

animais**.**extend(["cobra", 6]) *#Adiciona os elementos "cobra" e "6" para a lista*

animais

['papagaio', 12345, 6.5, ['leão', 'Cachorro'], 'cobra', 6]

animais**.**count("leão") *#Mostra quantas vezes apareceu a palavra "leão"*

lista**.**sort() *#Cria uma lista randomica*

lista

[10, 30, 80, 300, 500]

**Tuplas**

tp **=** ("Banana", "Maçã", 10, 50) *#As tuplas usam parênteses como sintaxe*

tp

('Banana', 'Maçã', 10, 50)

tp[0] *#Retornando o primeiro elemento*

'Banana'

*#Diferente das listas as tuplas são imutáveis, o que quer dizer que não podemos alterar os seus elementos*

*#Por isso o código abaixo da erro, se executar*

*#tp[0] = "Laranja"*

tp**.**count("Maçã") *#Mostra quantas vezes a palavra "Maçã" apareceu na tupla*

1

tp[0:2] *#Mostra os elementos do indice 0 até o valor depois dos dois pontos -1, ou seja. neste caso 0 até 1*

('Banana', 'Maçã')

**Dicionários**

*#Para criar um dicionário utilizamos as {}*

dc **=** {"Maçã":20, "Banana":10, "Laranja":15, "Uva":5} *#Dicionários trabalham com o conceito de chave e valor*

dc

{'Maçã': 20, 'Banana': 10, 'Laranja': 15, 'Uva': 5}

dc["Maçã"] *#Acessando o valor de um dicionário através da chave*

20

dc["Maçã"] **=** 25 *#Atualizando o valor da Maçã*

dc

{'Maçã': 25, 'Banana': 10, 'Laranja': 15, 'Uva': 5}

dc**.**keys() *#Retornando todas as chaves do dicionário*

dict\_keys(['Maçã', 'Banana', 'Laranja', 'Uva'])

dc**.**values() *#Retornando os valores do dicionário*

dict\_values([25, 10, 15, 5])

dc**.**setdefault("Limão", 22) *#Verificando se já existe uma chave no dicionário e caso não exista inserir*

22

dc

{'Maçã': 25, 'Banana': 10, 'Laranja': 15, 'Uva': 5, 'Limão': 22}

**Python para análise de dados (Pandas)**

|  | **country** | **continent** | **year** | **lifeExp** | **pop** | **gdpPercap** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | Afghanistan | Asia | 1952 | 28.80 | 8425333 | 779.45 |
| **1** | Afghanistan | Asia | 1957 | 30.33 | 9240934 | 820.85 |
| **2** | Afghanistan | Asia | 1962 | 32.00 | 10267083 | 853.10 |
| **3** | Afghanistan | Asia | 1967 | 34.02 | 11537966 | 836.20 |
| **4** | Afghanistan | Asia | 1972 | 36.09 | 13079460 | 739.98 |

*#Renomeando a tabela*

df **=** df**.**rename(columns**=**{"country":"Pais", "continent": "continente", "year":"Ano", "lifeExp":"Expectativa de vida", "pop":"Pop Total", "gdpPercap": "PIB"})

df**.**head(10)

|  | **Pais** | **continente** | **Ano** | **Expectativa de vida** | **Pop Total** | **PIB** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | Afghanistan | Asia | 1952 | 28.80 | 8425333 | 779.45 |
| **1** | Afghanistan | Asia | 1957 | 30.33 | 9240934 | 820.85 |
| **2** | Afghanistan | Asia | 1962 | 32.00 | 10267083 | 853.10 |
| **3** | Afghanistan | Asia | 1967 | 34.02 | 11537966 | 836.20 |
| **4** | Afghanistan | Asia | 1972 | 36.09 | 13079460 | 739.98 |
| **5** | Afghanistan | Asia | 1977 | 38.44 | 14880372 | 786.11 |
| **6** | Afghanistan | Asia | 1982 | 39.85 | 12881816 | 978.01 |
| **7** | Afghanistan | Asia | 1987 | 40.82 | 13867957 | 852.40 |
| **8** | Afghanistan | Asia | 1992 | 41.67 | 16317921 | 649.34 |
| **9** | Afghanistan | Asia | 1997 | 41.76 | 22227415 | 635.34 |

df**.**shape *#Total de linhas e colunas*

(3312, 6)

df**.**columns

Index(['Pais', 'continente', 'Ano', 'Expectativa de vida', 'Pop Total', 'PIB'], dtype='object')

df**.**dtypes

Pais object

continente object

Ano int64

Expectativa de vida float64

Pop Total int64

PIB float64

dtype: object

df**.**tail(15) *#As últimas 15 linhas*

|  | **Pais** | **continente** | **Ano** | **Expectativa de vida** | **Pop Total** | **PIB** |
| --- | --- | --- | --- | --- | --- | --- |
| **3297** | Zambia | Africa | 1997 | 40.24 | 9417789 | 1,071.35 |
| **3298** | Zambia | Africa | 2002 | 39.19 | 10595811 | 1,071.61 |
| **3299** | Zambia | Africa | 2007 | 42.38 | 11746035 | 1,271.21 |
| **3300** | Zimbabwe | Africa | 1952 | 48.45 | 3080907 | 406.88 |
| **3301** | Zimbabwe | Africa | 1957 | 50.47 | 3646340 | 518.76 |
| **3302** | Zimbabwe | Africa | 1962 | 52.36 | 4277736 | 527.27 |
| **3303** | Zimbabwe | Africa | 1967 | 53.99 | 4995432 | 569.80 |
| **3304** | Zimbabwe | Africa | 1972 | 55.63 | 5861135 | 799.36 |
| **3305** | Zimbabwe | Africa | 1977 | 57.67 | 6642107 | 685.59 |
| **3306** | Zimbabwe | Africa | 1982 | 60.36 | 7636524 | 788.86 |
| **3307** | Zimbabwe | Africa | 1987 | 62.35 | 9216418 | 706.16 |
| **3308** | Zimbabwe | Africa | 1992 | 60.38 | 10704340 | 693.42 |
| **3309** | Zimbabwe | Africa | 1997 | 46.81 | 11404948 | 792.45 |
| **3310** | Zimbabwe | Africa | 2002 | 39.99 | 11926563 | 672.04 |
| **3311** | Zimbabwe | Africa | 2007 | 43.49 | 12311143 | 469.71 |

df**.**describe()

|  | **Ano** | **Expectativa de vida** | **Pop Total** | **PIB** |
| --- | --- | --- | --- | --- |
| **count** | 3,312.00 | 3,312.00 | 3,312.00 | 3,312.00 |
| **mean** | 1,980.30 | 65.25 | 31,614,890.82 | 11,317.12 |
| **std** | 16.93 | 11.77 | 104,119,342.89 | 11,369.14 |
| **min** | 1,950.00 | 23.60 | 59,412.00 | 241.17 |
| **25%** | 1,967.00 | 58.34 | 2,678,572.00 | 2,514.63 |
| **50%** | 1,982.00 | 69.61 | 7,557,218.50 | 7,838.51 |
| **75%** | 1,996.00 | 73.66 | 19,585,221.75 | 17,357.88 |
| **max** | 2,007.00 | 82.67 | 1,318,683,096.00 | 113,523.13 |

df["continente"]**.**unique() *#Verificando os continentes*

array(['Asia', 'Europe', 'Africa', 'Americas', nan, 'FSU', 'Oceania'],

dtype=object)

Oceania **=** df**.**loc[df["continente"] **==** "Oceania"] *#Criando um dataFrame com dados do continente Oceania*

Oceania**.**head()

|  | **Pais** | **continente** | **Ano** | **Expectativa de vida** | **Pop Total** | **PIB** |
| --- | --- | --- | --- | --- | --- | --- |
| **975** | Fiji | Oceania | 1962 | 56.49 | 421869 | 2,039.39 |
| **976** | Fiji | Oceania | 1967 | 58.61 | 485000 | 2,170.07 |
| **977** | Fiji | Oceania | 1972 | 60.64 | 544000 | 2,797.76 |
| **978** | Fiji | Oceania | 1977 | 62.67 | 599339 | 3,182.57 |
| **979** | Fiji | Oceania | 1982 | 64.68 | 658906 | 3,209.26 |

Oceania["continente"]**.**unique()

array(['Oceania'], dtype=object)

df**.**groupby("continente")["Pais"]**.**nunique() *#Verificando quantos paises existem em cada continente na nossa base de dados*

continente

Africa 51

Americas 25

Asia 41

Europe 35

FSU 6

Oceania 3

Name: Pais, dtype: int64

df**.**groupby("Ano")["Expectativa de vida"]**.**mean() *#Qual a expectativa de vida média de cada ano?*

Ano

1950 62.00

1951 65.90

1952 49.21

1953 66.67

1954 67.46

1955 67.81

1956 67.95

1957 51.61

1958 68.82

1959 68.23

1960 68.47

1961 68.86

1962 54.04

1963 69.60

1964 70.06

1965 70.26

1966 70.45

1967 56.26

1968 70.69

1969 70.65

1970 70.96

1971 71.10

1972 58.47

1973 71.50

1974 71.78

1975 71.94

1976 72.16

1977 60.43

1978 72.72

1979 73.02

1980 73.06

1981 73.34

1982 62.37

1983 73.79

1984 74.10

1985 74.11

1986 74.45

1987 63.98

1988 74.76

1989 74.92

1990 74.28

1991 74.37

1992 65.01

1993 74.32

1994 74.46

1995 74.55

1996 75.03

1997 65.87

1998 75.57

1999 75.70

2000 76.03

2001 76.26

2002 66.84

2003 76.59

2004 76.92

2005 76.72

2006 77.89

2007 67.87

Name: Expectativa de vida, dtype: float64

df["PIB"]**.**mean() *#Média do PIB*

11317.115804696648

df["PIB"]**.**sum() *#Soma de todos os valores de "PIB"*

37482287.5451553

**Trabalhando com Planilhas do Excel**

**import** pandas **as** pd *#Importando a biblioteca pandas*

*#Leitura dos arquivos*

df1 **=** pd**.**read\_excel("Aracaju.xlsx")

df2 **=** pd**.**read\_excel("Fortaleza.xlsx")

df3 **=** pd**.**read\_excel("Natal.xlsx")

df4 **=** pd**.**read\_excel("Recife.xlsx")

df5 **=** pd**.**read\_excel("Salvador.xlsx")

df **=** pd**.**concat([df1,df2,df3,df4,df5]) *#Juntando todos os arquivos em apenas um dataFrame*

df**.**head() *#Exibindo as 5 primeiras linhas*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** |
| --- | --- | --- | --- | --- | --- |
| **0** | Aracaju | 2018-01-01 | 142.00 | 1520 | 1 |
| **1** | Aracaju | 2018-01-01 | 14.21 | 1522 | 6 |
| **2** | Aracaju | 2018-01-01 | 71.55 | 1520 | 1 |
| **3** | Aracaju | 2018-01-01 | 3.01 | 1521 | 7 |
| **4** | Aracaju | 2018-01-01 | 24.51 | 1522 | 8 |

df**.**tail() *#Exibindo as 5 últimas linhas*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** |
| --- | --- | --- | --- | --- | --- |
| **235** | Salvador | 2019-01-03 | 41.84 | 1034 | 1 |
| **236** | Salvador | 2019-01-03 | 126.29 | 1035 | 3 |
| **237** | Salvador | 2019-01-03 | 38.06 | 1036 | 3 |
| **238** | Salvador | 2019-01-03 | 139.64 | 1035 | 1 |
| **239** | Salvador | 2019-01-03 | 161.41 | 1037 | 3 |

df**.**sample(5)

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** |
| --- | --- | --- | --- | --- | --- |
| **91** | Aracaju | 2018-01-01 | 200.85 | 1522 | 6 |
| **167** | Salvador | 2019-01-02 | 36.42 | 1037 | 1 |
| **45** | Fortaleza | 2019-01-01 | 39.63 | 1004 | 3 |
| **76** | Recife | 2019-01-01 | 17.86 | 980 | 5 |
| **219** | Natal | 2019-01-02 | 19.67 | 1037 | 2 |

df**.**dtypes *#Verificando o tipo de dado de cada coluna*

Cidade object

Data datetime64[ns]

Vendas float64

LojaID int64

Qtde int64

dtype: object

df["LojaID"] **=** df["LojaID"]**.**astype("object") *#Alterando o tipo de dado da coluna LojaID*

df**.**dtypes

Cidade object

Data datetime64[ns]

Vendas float64

LojaID object

Qtde int64

dtype: object

**Tratando valores faltantes**

df**.**isnull()**.**sum() *#Verificando linhas com valores faltantes*

Cidade 0

Data 0

Vendas 0

LojaID 0

Qtde 0

dtype: int64

df["Vendas"]**.**fillna(df["Vendas"]**.**mean(), inplace**=True**) *#Substituindo os valores nulos pela média*

df["Vendas"]**.**mean() *#Verificando a média da coluna "vendas" dataFrame*

122.61180089485458

df**.**describe()

|  | **Vendas** | **Qtde** |
| --- | --- | --- |
| **count** | 894.00 | 894.00 |
| **mean** | 122.61 | 3.26 |
| **std** | 166.82 | 2.10 |
| **min** | 3.01 | 1.00 |
| **25%** | 22.38 | 2.00 |
| **50%** | 44.00 | 3.00 |
| **75%** | 160.93 | 4.00 |
| **max** | 886.00 | 10.00 |

df**.**isnull()**.**sum()

Cidade 0

Data 0

Vendas 0

LojaID 0

Qtde 0

dtype: int64

df**.**sample(15) *#Amostra de 15 dados do dataSet*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** |
| --- | --- | --- | --- | --- | --- |
| **226** | Salvador | 2019-01-02 | 38.32 | 1035 | 2 |
| **100** | Salvador | 2019-01-01 | 190.97 | 1036 | 2 |
| **160** | Natal | 2019-01-02 | 11.76 | 1034 | 1 |
| **62** | Aracaju | 2018-01-01 | 28.66 | 1520 | 3 |
| **114** | Natal | 2019-01-02 | 25.52 | 1034 | 1 |
| **26** | Fortaleza | 2019-01-01 | 18.93 | 1005 | 2 |
| **188** | Natal | 2019-01-02 | 178.11 | 1035 | 3 |
| **43** | Salvador | 2019-01-01 | 12.21 | 1034 | 1 |
| **176** | Salvador | 2019-01-02 | 19.42 | 1035 | 3 |
| **173** | Salvador | 2019-01-02 | 31.77 | 1034 | 2 |
| **28** | Aracaju | 2018-01-01 | 114.93 | 1520 | 5 |
| **95** | Recife | 2019-01-01 | 40.03 | 982 | 3 |
| **25** | Fortaleza | 2019-01-01 | 80.31 | 1004 | 4 |
| **141** | Fortaleza | 2019-03-02 | 182.75 | 983 | 8 |
| **21** | Recife | 2019-01-01 | 164.74 | 983 | 1 |

df["Vendas"]**.**fillna(0, inplace**=True**) *#Substituindo os valores nulos por zero*

df**.**dropna(inplace**=True**) *#Apagando as linhas com valores nulos*

df**.**dropna(subset**=**["Vendas"], inplace**=True**) *#Apagando as linhas com valores nulos com base apenas em 1 coluna*

df**.**dropna(how**=**"all", inplace**=True**) *#Removendo linhas que estejam com valores faltantes em todas as colunas*

**Criando colunas novas**

df["Receita"] **=** df["Vendas"]**.**mul(df["Qtde"]) *#Criando a coluna de "receita", que pega o valor das "vendas" e multiplica pela "quantidade"*

df**.**head()

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | Aracaju | 2018-01-01 | 142.00 | 1520 | 1 | 142.00 |
| **1** | Aracaju | 2018-01-01 | 14.21 | 1522 | 6 | 85.26 |
| **2** | Aracaju | 2018-01-01 | 71.55 | 1520 | 1 | 71.55 |
| **3** | Aracaju | 2018-01-01 | 3.01 | 1521 | 7 | 21.07 |
| **4** | Aracaju | 2018-01-01 | 24.51 | 1522 | 8 | 196.08 |

df["Receita/Vendas"] **=** df["Receita"] **/** df["Vendas"] *#Criando a coluna de "Receitas/Vendas", que pega o valor das "Receitas" e divide pela "Vendas"*

df**.**head()

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Aracaju | 2018-01-01 | 142.00 | 1520 | 1 | 142.00 | 1.00 |
| **1** | Aracaju | 2018-01-01 | 14.21 | 1522 | 6 | 85.26 | 6.00 |
| **2** | Aracaju | 2018-01-01 | 71.55 | 1520 | 1 | 71.55 | 1.00 |
| **3** | Aracaju | 2018-01-01 | 3.01 | 1521 | 7 | 21.07 | 7.00 |
| **4** | Aracaju | 2018-01-01 | 24.51 | 1522 | 8 | 196.08 | 8.00 |

df["Receita"]**.**max() *#Retornando a maior receita*

3544.0

df["Receita"]**.**min() *#Retornando a menor receita*

3.34

df**.**nlargest(3, "Receita") *#Mostra o Top "3" com base na coluna "Receita" (maiores receitas)*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **7** | Natal | 2019-03-18 | 886.00 | 853 | 4 | 3,544.00 | 4.00 |
| **51** | Natal | 2018-01-21 | 859.00 | 852 | 4 | 3,436.00 | 4.00 |
| **55** | Natal | 2019-01-08 | 859.00 | 854 | 4 | 3,436.00 | 4.00 |

df**.**nsmallest(3, "Receita") *#Retorna o Top 3, com as piores "Receitas"*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **118** | Aracaju | 2018-01-01 | 3.34 | 1522 | 1 | 3.34 | 1.00 |
| **65** | Recife | 2019-01-01 | 4.01 | 981 | 1 | 4.01 | 1.00 |
| **92** | Natal | 2019-01-02 | 4.57 | 1035 | 1 | 4.57 | 1.00 |

df**.**groupby("Cidade")["Receita"]**.**sum() *#Agrupamento por cidade*

Cidade

Aracaju 48,748.25

Fortaleza 37,913.97

Natal 167,227.52

Recife 51,936.51

Salvador 40,596.73

Name: Receita, dtype: float64

df**.**sort\_values("Receita", ascending**=False**)**.**head(10) *#Ordenando o conjunto de dados com as maiores receitas, do maior para o menor*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **7** | Natal | 2019-03-18 | 886.00 | 853 | 4 | 3,544.00 | 4.00 |
| **55** | Natal | 2019-01-08 | 859.00 | 854 | 4 | 3,436.00 | 4.00 |
| **51** | Natal | 2018-01-21 | 859.00 | 852 | 4 | 3,436.00 | 4.00 |
| **30** | Natal | 2018-10-02 | 856.00 | 853 | 4 | 3,424.00 | 4.00 |
| **41** | Natal | 2018-05-20 | 835.00 | 852 | 4 | 3,340.00 | 4.00 |
| **38** | Natal | 2018-02-25 | 828.00 | 852 | 4 | 3,312.00 | 4.00 |
| **10** | Natal | 2018-10-27 | 828.00 | 852 | 4 | 3,312.00 | 4.00 |
| **69** | Natal | 2019-03-24 | 817.00 | 852 | 4 | 3,268.00 | 4.00 |
| **62** | Natal | 2018-02-10 | 793.00 | 854 | 4 | 3,172.00 | 4.00 |
| **52** | Natal | 2018-04-27 | 778.00 | 854 | 4 | 3,112.00 | 4.00 |

**Trabalhando com datas**

Cidade object

Data int64

Vendas float64

LojaID object

Qtde int64

Receita float64

Receita/Vendas float64

dtype: object

df["Data"] **=** pd**.**to\_datetime(df["Data"]) *#Transformando coluna de data no tipo "data"*

df**.**dtypes

Cidade object

Data datetime64[ns]

Vendas float64

LojaID object

Qtde int64

Receita float64

Receita/Vendas float64

dtype: object

df**.**groupby(df["Data"]**.**dt**.**year)["Receita"]**.**sum() *#Receita por ano*

Data

2018 118,176.53

2019 228,246.45

Name: Receita, dtype: float64

df["Ano\_Venda"] **=** df["Data"]**.**dt**.**year *#Criando uma nova coluna com o ano*

df**.**sample(5) *#Pega uma amostra do dataSet*

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** | **Ano\_Venda** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **100** | Recife | 2019-01-01 | 35.11 | 982 | 4 | 140.44 | 4.00 | 2019 |
| **213** | Salvador | 2019-01-02 | 42.12 | 1035 | 1 | 42.12 | 1.00 | 2019 |
| **134** | Fortaleza | 2019-03-02 | 195.18 | 980 | 5 | 975.90 | 5.00 | 2019 |
| **43** | Recife | 2019-01-01 | 22.29 | 983 | 7 | 156.03 | 7.00 | 2019 |
| **1** | Aracaju | 2018-01-01 | 14.21 | 1522 | 6 | 85.26 | 6.00 | 2018 |

df["mes\_venda"], df["dia\_venda"] **=** (df["Data"]**.**dt**.**month, df["Data"]**.**dt**.**day) *#Extraindo o mês e o dia*

df**.**sample(5)

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** | **Ano\_Venda** | **mes\_venda** | **dia\_venda** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **167** | Salvador | 2019-01-02 | 36.42 | 1037 | 1 | 36.42 | 1.00 | 2019 | 1 | 2 |
| **21** | Recife | 2019-01-01 | 164.74 | 983 | 1 | 164.74 | 1.00 | 2019 | 1 | 1 |
| **73** | Salvador | 2019-01-01 | 8.39 | 1037 | 1 | 8.39 | 1.00 | 2019 | 1 | 1 |
| **71** | Recife | 2019-01-01 | 138.69 | 983 | 3 | 416.07 | 3.00 | 2019 | 1 | 1 |
| **43** | Salvador | 2019-01-01 | 12.21 | 1034 | 1 | 12.21 | 1.00 | 2019 | 1 | 1 |

df["Data"]**.**min() *#Retornando a data mais antiga*

Timestamp('2018-01-01 00:00:00')

df["diferenca\_dias"] **=** df["Data"] **-** df["Data"]**.**min() *#Calculando a diferença de dias*

df**.**sample(5)

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** | **Ano\_Venda** | **mes\_venda** | **dia\_venda** | **diferenca\_dias** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **110** | Fortaleza | 2019-03-02 | 51.98 | 983 | 6 | 311.88 | 6.00 | 2019 | 3 | 2 | 425 days |
| **212** | Natal | 2019-01-02 | 50.73 | 1037 | 3 | 152.19 | 3.00 | 2019 | 1 | 2 | 366 days |
| **5** | Salvador | 2019-01-01 | 179.20 | 1037 | 3 | 537.60 | 3.00 | 2019 | 1 | 1 | 365 days |
| **126** | Salvador | 2019-03-02 | 41.69 | 1036 | 2 | 83.38 | 2.00 | 2019 | 3 | 2 | 425 days |
| **52** | Fortaleza | 2019-01-01 | 15.22 | 1003 | 2 | 30.44 | 2.00 | 2019 | 1 | 1 | 365 days |

df["trimestre\_venda"] **=** df["Data"]**.**dt**.**quarter *#Criando a coluna de trimestre*

df**.**sample(5)

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** | **Ano\_Venda** | **mes\_venda** | **dia\_venda** | **diferenca\_dias** | **trimestre\_venda** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | Aracaju | 2018-01-01 | 37.49 | 1521 | 2 | 74.98 | 2.00 | 2018 | 1 | 1 | 0 days | 1 |
| **36** | Aracaju | 2018-01-01 | 37.99 | 1523 | 8 | 303.92 | 8.00 | 2018 | 1 | 1 | 0 days | 1 |
| **65** | Aracaju | 2018-01-01 | 145.54 | 1522 | 3 | 436.62 | 3.00 | 2018 | 1 | 1 | 0 days | 1 |
| **107** | Natal | 2019-01-02 | 14.70 | 1035 | 1 | 14.70 | 1.00 | 2019 | 1 | 2 | 366 days | 1 |
| **10** | Natal | 2018-10-27 | 828.00 | 852 | 4 | 3,312.00 | 4.00 | 2018 | 10 | 27 | 299 days | 4 |

vendas\_marco\_19 **=** df**.**loc[(df["Data"]**.**dt**.**year **==** 2019) **&** (df["Data"]**.**dt**.**month **==** 3)] *#Filtrando as vendas de 2019 do mês de março*

vendas\_marco\_19**.**sample(20)

|  | **Cidade** | **Data** | **Vendas** | **LojaID** | **Qtde** | **Receita** | **Receita/Vendas** | **Ano\_Venda** | **mes\_venda** | **dia\_venda** | **diferenca\_dias** | **trimestre\_venda** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **108** | Recife | 2019-03-02 | 152.89 | 981 | 4 | 611.56 | 4.00 | 2019 | 3 | 2 | 425 days | 1 |
| **121** | Recife | 2019-03-02 | 41.55 | 981 | 3 | 124.65 | 3.00 | 2019 | 3 | 2 | 425 days | 1 |
| **138** | Recife | 2019-03-02 | 150.38 | 983 | 6 | 902.28 | 6.00 | 2019 | 3 | 2 | 425 days | 1 |
| **112** | Salvador | 2019-03-02 | 43.00 | 1036 | 3 | 129.00 | 3.00 | 2019 | 3 | 2 | 425 days | 1 |
| **136** | Salvador | 2019-03-02 | 13.81 | 1036 | 1 | 13.81 | 1.00 | 2019 | 3 | 2 | 425 days | 1 |
| **115** | Fortaleza | 2019-03-02 | 12.23 | 981 | 3 | 36.69 | 3.00 | 2019 | 3 | 2 | 425 days | 1 |
| **125** | Salvador | 2019-03-02 | 25.00 | 1037 | 2 | 50.00 | 2.00 | 2019 | 3 | 2 | 425 days | 1 |
| **111** | Salvador | 2019-03-02 | 147.35 | 1037 | 2 | 294.70 | 2.00 | 2019 | 3 | 2 | 425 days | 1 |
| **125** | Fortaleza | 2019-03-02 | 37.60 | 983 | 4 | 150.40 | 4.00 | 2019 | 3 | 2 | 425 days | 1 |
| **116** | Recife | 2019-03-02 | 79.10 | 982 | 2 | 158.20 | 2.00 | 2019 | 3 | 2 | 425 days | 1 |
| **132** | Fortaleza | 2019-03-02 | 17.49 | 983 | 5 | 87.45 | 5.00 | 2019 | 3 | 2 | 425 days | 1 |
| **128** | Fortaleza | 2019-03-02 | 38.70 | 982 | 4 | 154.80 | 4.00 | 2019 | 3 | 2 | 425 days | 1 |
| **2** | Natal | 2019-03-11 | 308.00 | 852 | 3 | 924.00 | 3.00 | 2019 | 3 | 11 | 434 days | 1 |
| **127** | Recife | 2019-03-02 | 39.56 | 982 | 8 | 316.48 | 8.00 | 2019 | 3 | 2 | 425 days | 1 |
| **71** | Natal | 2019-03-20 | 840.00 | 854 | 3 | 2,520.00 | 3.00 | 2019 | 3 | 20 | 443 days | 1 |
| **118** | Fortaleza | 2019-03-02 | 17.70 | 981 | 5 | 88.50 | 5.00 | 2019 | 3 | 2 | 425 days | 1 |
| **120** | Salvador | 2019-03-02 | 140.17 | 1036 | 2 | 280.34 | 2.00 | 2019 | 3 | 2 | 425 days | 1 |
| **137** | Salvador | 2019-03-02 | 51.66 | 1036 | 3 | 154.98 | 3.00 | 2019 | 3 | 2 | 425 days | 1 |
| **108** | Fortaleza | 2019-03-02 | 152.89 | 981 | 4 | 611.56 | 4.00 | 2019 | 3 | 2 | 425 days | 1 |
| **42** | Natal | 2019-03-02 | 775.00 | 854 | 2 | 1,550.00 | 2.00 | 2019 | 3 | 2 | 425 days | 1 |

**Visualização de dados**

df["LojaID"]**.**value\_counts(ascending**=False**) *#Ordena as vendas pelo ID das lojas e organiza do maior para o menor*

1036 117

1035 112

1037 101

1034 67

983 65

982 44

1522 41

1520 39

980 36

981 31

1005 30

1002 30

852 29

1523 29

1004 28

854 28

853 26

1521 21

1003 20

Name: LojaID, dtype: int64

df["LojaID"]**.**value\_counts(ascending**=False**)**.**plot**.**bar() *#Gráfico de barras*

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f1ccd0916d0>

df["LojaID"]**.**value\_counts()**.**plot**.**barh() *#Gráfico de barras horizontais*

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f1ccd04a690>

df["LojaID"]**.**value\_counts(ascending**=True**)**.**plot**.**barh(); *#Gráfico de barras horizontais*

df**.**groupby(df["Data"]**.**dt**.**year)["Receita"]**.**sum()**.**plot**.**pie() *#Gráfico de Pizza*

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f1ccced6910>

df["Cidade"]**.**value\_counts() *#Total de vendas por cidade*

Natal 240

Salvador 240

Fortaleza 142

Recife 142

Aracaju 130

Name: Cidade, dtype: int64

*#Adicionando um título e alterando o nome aos eixos*

**import** matplotlib.pyplot **as** plt

df["Cidade"]**.**value\_counts()**.**plot**.**bar(title**=**"Total vendas por Cidade", color**=**"lightpink")

plt**.**xlabel("Cidade")

plt**.**ylabel("Total Vendas");

*#Alterando a cor*

df["Cidade"]**.**value\_counts()**.**plot**.**bar(title**=**"Total vendas por Cidade", color**=**['tan', 'red', 'lightpink', 'blue', 'cyan'])

plt**.**xlabel("Cidade")

plt**.**ylabel("Total Vendas");

*#Alterando o estilo*

plt**.**style**.**use("ggplot")

df**.**groupby(df["mes\_venda"])["Qtde"]**.**sum()**.**plot(title **=** "Total Produtos vendidos x mês")

plt**.**xlabel("Mês")

plt**.**ylabel("Total Produtos Vendidos")

plt**.**legend();

df**.**groupby(df["mes\_venda"])["Qtde"]**.**sum()

mes\_venda

1 2208

2 144

3 467

4 23

5 17

6 13

7 13

8 2

9 10

10 14

11 2

12 3

Name: Qtde, dtype: int64

df\_2019 **=** df[df["Ano\_Venda"] **==** 2019] *#Selecionando apenas as vendas de 2019*

df\_2019**.**groupby(df\_2019["mes\_venda"])["Qtde"]**.**sum()

mes\_venda

1 1541

2 128

3 460

4 12

Name: Qtde, dtype: int64

*#Total produtos vendidos por mês*

df\_2019**.**groupby(df\_2019["mes\_venda"])["Qtde"]**.**sum()**.**plot(marker **=** "o")

plt**.**xlabel("Mês")

plt**.**ylabel("Total Produtos Vendidos")

plt**.**legend();

plt**.**hist(df["Qtde"], color**=**"lightpink"); *#Hisograma*

plt**.**scatter(x**=**df\_2019["dia\_venda"], y **=** df\_2019["Receita"]); *#Gráfico de dispersão*

df\_2019**.**groupby(df\_2019["mes\_venda"])["Qtde"]**.**sum()**.**plot(marker **=** "v")

plt**.**title("Quantidade de produtos vendidos x mês")

plt**.**xlabel("Mês")

plt**.**ylabel("Total Produtos Vendidos");

plt**.**legend()

plt**.**savefig("grafico QTDE x MES.png") *#Salvando em png*

**Análise exploratória**

*#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()

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving AdventureWorks.xlsx to AdventureWorks (2).xlsx

df **=** pd**.**read\_excel("AdventureWorks.xlsx") *#Criando nosso DataFrame*

df**.**head() *#Visualizando as 5 primeiras linhas*

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red |
| **1** | 2008-05-12 | 2008-05-17 | 306 | 384 | 19051 | 200805123CS567 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red |
| **2** | 2008-05-14 | 2008-05-20 | 306 | 384 | 19052 | 200805143CS576 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red |
| **3** | 2008-05-21 | 2008-05-27 | 306 | 384 | 19052 | 200805213CS576 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red |
| **4** | 2008-06-20 | 2008-06-27 | 306 | 384 | 19053 | 200806203CS586 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red |

df**.**shape *#Quantidade de linhas e colunas*

(904, 16)

df**.**dtypes *#Verificando os tipos de dados*

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

df["Valor Venda"]**.**sum() *#Qual a Receita total?*

5984606.1426

*#Qual o custo Total?*

df["custo"] **=** df["Custo Unitário"]**.**mul(df["Quantidade"]) *#Criando a coluna de custo = custo unitário x quantidade*

df**.**head(1)

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 |

round(df["custo"]**.**sum(), 2) *#Qual o custo Total?*

2486783.05

**Agora que temos a receita e custo e o total, podemos achar o Lucro total.**

Lucro = Valor da venda - custo

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

df**.**head(1)

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** | **lucro** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 |

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

3497823.09

df["Tempo\_envio"] **=** df["Data Envio"] **-** df["Data Venda"] *#Criando uma coluna com total de dias para enviar o produto*

df**.**head(1)

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** | **lucro** | **Tempo\_envio** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 20 days |

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

df["Tempo\_envio"] **=** (df["Data Envio"] **-** df["Data Venda"])**.**dt**.**days *#Extraindo apenas os dias*

df**.**head(1)

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** | **lucro** | **Tempo\_envio** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 20 |

df["Tempo\_envio"]**.**dtype *#Verificando o tipo da coluna Tempo\_envio*

dtype('int64')

df**.**groupby("Marca")["Tempo\_envio"]**.**mean() *#Média do tempo de envio por Marca*

Marca

Adventure Works 8.66

Contoso 8.47

Fabrikam 8.51

Name: Tempo\_envio, dtype: float64

**Missing Values (valores faltantes)**

df**.**isnull()**.**sum() *#Verificando se temos dados faltantes*

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

**E, se a gente quiser saber o Lucro por Ano e Por Marca?**

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

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

*#Resetando o index*

lucro\_ano **=** df**.**groupby([df["Data Venda"]**.**dt**.**year, "Marca"])["lucro"]**.**sum()**.**reset\_index()

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 |

df**.**groupby("Produto")["Quantidade"]**.**sum()**.**sort\_values(ascending**=False**) *#Qual o total de produtos vendidos?*

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

*#Gráfico Total de produtos vendidos*

df**.**groupby("Produto")["Quantidade"]**.**sum()**.**sort\_values(ascending**=True**)**.**plot**.**barh(title**=**"Total Produtos Vendidos", color**=**"lightpink")

plt**.**xlabel("Total")

plt**.**ylabel("Produto");

df**.**groupby(df["Data Venda"]**.**dt**.**year)["lucro"]**.**sum()**.**plot**.**bar(title**=**"Lucro x Ano", color**=**"lightpink")

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** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** | **lucro** | **Tempo\_envio** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **11** | 2009-05-02 | 2009-05-14 | 199 | 384 | 18938 | 200905023CS847 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 12 |
| **12** | 2009-05-16 | 2009-05-27 | 306 | 384 | 19067 | 200905163CS746 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 11 |
| **13** | 2009-05-31 | 2009-06-12 | 306 | 384 | 19068 | 200905313CS748 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 12 |
| **14** | 2009-06-28 | 2009-07-11 | 199 | 384 | 18959 | 200906283CS870 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 13 |
| **15** | 2009-07-08 | 2009-07-12 | 306 | 384 | 19070 | 200907083CS757 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 4 |

df\_2009**.**groupby(df\_2009["Data Venda"]**.**dt**.**month)["lucro"]**.**sum()**.**plot(title**=**"Lucro x Mês")

plt**.**xlabel("Mês")

plt**.**ylabel("Lucro");

df\_2009**.**groupby("Marca")["lucro"]**.**sum()**.**plot**.**bar(title**=**"Lucro x Marca", color**=**"lightpink")

plt**.**xlabel("Marca")

plt**.**ylabel("Lucro")

plt**.**xticks(rotation**=**'horizontal');

df\_2009**.**groupby("Classe")["lucro"]**.**sum()**.**plot**.**bar(title**=**"Lucro x Classe", color**=**"lightpink")

plt**.**xlabel("Classe")

plt**.**ylabel("Lucro")

plt**.**xticks(rotation**=**'horizontal');

df["Tempo\_envio"]**.**describe()

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

plt**.**boxplot(df["Tempo\_envio"]); *#Gráfico de Boxplot*

plt**.**hist(df["Tempo\_envio"]); *#Histograma*

df["Tempo\_envio"]**.**min() *#Tempo mínimo de envio*

4

df['Tempo\_envio']**.**max() *#Tempo máximo de envio*

20

df[df["Tempo\_envio"] **==** 20] *#Identificando o Outlier (valor discrepante)*

|  | **Data Venda** | **Data Envio** | **ID Loja** | **ID Produto** | **ID Cliente** | **No. Venda** | **Custo Unitário** | **Preço Unitário** | **Quantidade** | **Valor Desconto** | **Valor Venda** | **Produto** | **Fabricante** | **Marca** | **Classe** | **Cor** | **custo** | **lucro** | **Tempo\_envio** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 2008-05-09 | 2008-05-29 | 199 | 384 | 18839 | 200805093CS607 | 348.58 | 758.00 | 6 | 0.00 | 4,548.00 | Adventure Works Laptop15.4W M1548 Red | Adventure Works | Adventure Works | Regular | Red | 2,091.48 | 2,456.52 | 20 |

df**.**to\_csv("df\_vendas\_novo.csv", index**=False**) *#Exportando os dados para um arquivo CSV*