Análise dos Dados para empresa M.Soluções

Arquivo: Vendas(1).xlsx

Tipo de modelo de Negócio: E-commerce

Requisitos de Negócio:
-Análise das Vendas;
-Criar uma loja física em uma Região;
-Estudar o Público da Região;

In [1]:
```
# Importando bibliotecas

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
plt.style.use("ggplot")
import colorsys
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
import matplotlib.pyplot as plt
import folium
```

In [2]:
```
# Chamando o arquivo
df = pd.read_excel("Vendas (1).xlsx")
```

Analisando os dados da Planilha

In [3]:
```
# Verificando as primeiras linhas do data frame
df.head()
```

Out[3]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continente	Unnamed: 12
0	2017-06-01 00:00:00	Home Theater System 7.1 Channel X711 Silver	Home Theater System	1109.00	367.43	Litware	1.0	1109.00	Levi	Rana	França	Europa	NaN
1	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Heidi	Patel	Estados Unidos	América do Norte	NaN
2	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Martin	Gonzalez	Chile	América do Sul	NaN
3	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Victor	Ruiz	Alemanha	Europa	NaN
4	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	5.0	1078.10	Alex	Nelson	Estados Unidos	América do Norte	NaN

In [4]:
```
### Análise dos dados acima, foi encontrado problemas, tais como:
###- colunas em branco
###- colunas que não são necessárias para análise
```

In [5]:
```
# Verificando quantidades de linhas e colunas do Data frame
df.shape
```

Out[5]:
```
(203888, 13)
```

In [6]:
```
# Verificando quantidades de registro por cada coluna(detalhadamente)
df.count()
```

Out[6]:
```
Data da Venda     203888
Produto           203882
Subcategoria      203882
PrecoUnitario     203882
Custo Unitário    203882
Marca             203882
Qtd. Vendida      203882
Faturamento       203882
Nome Cliente      203882
Sobrenome         203882
País              203882
Continente        203882
Unnamed: 12            2
dtype: int64
```

In [7]:
```
# Análise matemática das colunas númericas
df.describe()
```

Out[7]:

	PrecoUnitario	Custo Unitário	Qtd. Vendida	Faturamento
count	203882.000000	203882.000000	203882.000000	203882.000000
mean	159.804995	66.437852	1.986880	314.754825
std	224.677022	85.880331	1.408089	583.330208
min	4.980000	2.540000	1.000000	4.980000
25%	8.990000	4.130000	1.000000	21.560000
50%	70.130000	32.250000	1.000000	99.990000
75%	181.000000	82.320000	3.000000	329.000000
max	1650.000000	546.680000	5.000000	8250.000000

In [8]:
```
# Verificando os tipos de dados de cada coluna
df.dtypes
```

Out[8]:
```
Data da Venda      object
Produto            object
Subcategoria       object
PrecoUnitario     float64
Custo Unitário    float64
Marca              object
Qtd. Vendida      float64
Faturamento       float64
Nome Cliente       object
Sobrenome          object
País               object
Continente         object
Unnamed: 12        object
dtype: object
```

In [9]:
```
### Análise dos dados acima, foi encontrado problema, tais como:
# - Data da venda, está como object e deve estar "Date"
```

In [10]:
```
# Verificando colunas dados em branco
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 203888 entries, 0 to 203887
Data columns (total 13 columns):
 #   Column          Non-Null Count   Dtype  
---  ------          --------------   -----  
 0   Data da Venda   203888 non-null  object 
 1   Produto         203882 non-null  object 
 2   Subcategoria    203882 non-null  object 
 3   PrecoUnitario   203882 non-null  float64
 4   Custo Unitário  203882 non-null  float64
 5   Marca           203882 non-null  object 
 6   Qtd. Vendida    203882 non-null  float64
 7   Faturamento     203882 non-null  float64
 8   Nome Cliente    203882 non-null  object 
 9   Sobrenome       203882 non-null  object 
 10  País            203882 non-null  object 
 11  Continente      203882 non-null  object 
 12  Unnamed: 12     2 non-null       object 
dtypes: float64(4), object(9)
memory usage: 20.2+ MB
```

In [11]:
```
# Verificando as ultimas linhas do data frame,
df.tail(10)
```

Out[11]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continente	U
203878	2019-08-31 00:00:00	Wireless Bluetooth Stereo Headphones M402 Green	Bluetooth Headphones	99.99	45.98	Northwind Traders	5.0	499.95	Mary	Adams	Chile	América do Sul	
203879	2019-08-31 00:00:00	Wireless Bluetooth Stereo Headphones M402 Green	Bluetooth Headphones	99.99	45.98	Northwind Traders	3.0	299.97	Morgan	Rivera	Estados Unidos	América do Norte	
203880	2019-08-31 00:00:00	Wireless Bluetooth Stereo Headphones M402 Green	Bluetooth Headphones	99.99	45.98	Northwind Traders	1.0	99.99	Adam	Kumar	Estados Unidos	América do Norte	
203881	2019-08-31 00:00:00	Wireless Bluetooth Stereo Headphones M402 Red	Bluetooth Headphones	99.99	45.98	Northwind Traders	3.0	299.97	Fernando	Rodriguez	Estados Unidos	América do Norte	
203882		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
203883		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
203884		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
203885		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
203886		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
203887		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

In [12]:
```
### Análise dos dados acima, foi encontrado problema, tais como:
# - linhas em branco
```

In [13]:
```
# Verificando o problema da coluna Data da venda
df["Data da Venda"].value_counts()
```

Out[13]:
```
2017-11-12 00:00:00    939
2017-11-11 00:00:00    926
2017-11-19 00:00:00    890
2017-11-21 00:00:00    864
2017-11-13 00:00:00    786
                      ...
2017-09-06 00:00:00    103
2017-10-15 00:00:00     87
2019-08-04 00:00:00     84
2017-09-15 00:00:00     79
                        6
Name: Data da Venda, Length: 823, dtype: int64
```

In [14]:
```
### Análise dos dados acima, foi encontrado problema, tais como:
# - Data da venda, tem horas
```

In [15]:
```
# Verificando em outro formato colunas e linhas
df.head(3).T
```

Out[15]:

	0	1	2
Data da Venda	2017-06-01 00:00:00	2017-06-01 00:00:00	2017-06-01 00:00:00
Produto	Home Theater System 7.1 Channel X711 Silver	180 CFM Vertical Discharge Fan X450 Black	180 CFM Vertical Discharge Fan X450 Black
Subcategoria	Home Theater System	Fans	Fans
PrecoUnitario	1109	215.62	215.62
Custo Unitário	367.43	71.44	71.44
Marca	Litware	Litware	Litware
Qtd. Vendida	1	1	1
Faturamento	1109	215.62	215.62
Nome Cliente	Levi	Heidi	Martin
Sobrenome	Rana	Patel	Gonzalez
País	França	Estados Unidos	Chile
Continente	Europa	América do Norte	América do Sul
Unnamed: 12	NaN	NaN	NaN

In [16]:
```
# Verificando os dados por tipo
# df.select_dtypes("object").head()
```

Manipulação dos Dados

In [17]:
```
# Resumo de dados a serem Manipulados
# 1 - colunas em branco; **********************************************************************resolvido
# 2 - colunas que não são necessárias para análise;*****************************************não será preciso
# 3 - Data da venda, está como object e deve estar "Date";**********************************não será preciso
# 4 - linhas em branco;************************************************************************resolvido
# 5 - "Data da venda", tem incluso formato horas.******************************************resolvido
```

In [18]:
```
# Copiando um data frame do original e criando data frame " df_novo"
df_novo = df.copy()
```

1 - Colunas e Linhas em branco;

In [19]:
```

# OBS:Comando para excluir o NaN das colunas e linhas - df_novo.dropna()

# Somando Total dos valores faltantes
df_novo.isnull().sum()
```

Out[19]:
```
Data da Venda          0
Produto                6
Subcategoria           6
PrecoUnitario          6
Custo Unitário         6
Marca                  6
Qtd. Vendida           6
Faturamento            6
Nome Cliente           6
Sobrenome              6
País                   6
Continente             6
Unnamed: 12       203886
dtype: int64
```

In [20]:
```
# Preenchendo os dados faltantes ou vazios por falso ou verdadeiro
enulo = df_novo.isnull()
```

In [21]:
```
# Verificando as primeiras linhas
enulo.head(7)
```

Out[21]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continente	Unnamed: 12
0	False	False	False	False	False	False	False	False	False	False	False	False	True
1	False	False	False	False	False	False	False	False	False	False	False	False	True
2	False	False	False	False	False	False	False	False	False	False	False	False	True
3	False	False	False	False	False	False	False	False	False	False	False	False	True
4	False	False	False	False	False	False	False	False	False	False	False	False	True
5	False	False	False	False	False	False	False	False	False	False	False	False	True
6	False	False	False	False	False	False	False	False	False	False	False	False	True

In [22]:
```
# Verificando as ultimas linhas
enulo.tail(7)
```

Out[22]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continente	Unnamed: 12
203881	False	False	False	False	False	False	False	False	False	False	False	False	True
203882	False	True	True	True	True	True	True	True	True	True	True	True	True
203883	False	True	True	True	True	True	True	True	True	True	True	True	True
203884	False	True	True	True	True	True	True	True	True	True	True	True	True
203885	False	True	True	True	True	True	True	True	True	True	True	True	True
203886	False	True	True	True	True	True	True	True	True	True	True	True	True
203887	False	True	True	True	True	True	True	True	True	True	True	True	True

In [23]:
```
# Criando uma váriavel com a soma dos valores verdadeiros
faltantes = df_novo.isnull().sum()
```

In [24]:
```
# Verificando os dados acima
print(faltantes)
```

```
Data da Venda          0
Produto                6
Subcategoria           6
PrecoUnitario          6
Custo Unitário         6
Marca                  6
Qtd. Vendida           6
Faturamento            6
Nome Cliente           6
Sobrenome              6
País                   6
Continente             6
Unnamed: 12       203886
dtype: int64
```

In [25]:
```
# Calculando em porcentagem
# faltantes% = (df_novo.isnull().sum() / len(df_novo["coluna que mostra o comprimento do dataset"]))*100
```

In [26]:
```
# Preenchendo os valores faltantes das string por "corrigido" e levantando a media dos valores
df_novo["Produto"].fillna("corrigido", inplace = True)
df_novo["Subcategoria"].fillna("corrigido", inplace = True)
df_novo["PrecoUnitario"].fillna(df_novo["PrecoUnitario"].mean(), inplace=True)
df_novo["Custo Unitário"].fillna(df_novo["Custo Unitário"].mean(), inplace=True)
df_novo["Marca"].fillna("corrigido", inplace = True)
df_novo["Qtd. Vendida"].fillna(df_novo["Custo Unitário"].mean(), inplace=True)
df_novo["Faturamento"].fillna(df_novo["Faturamento"].mean(), inplace=True)
df_novo["Nome Cliente"].fillna("corrigido", inplace = True)
df_novo["Sobrenome"].fillna("corrigido", inplace = True)
df_novo["País"].fillna("corrigido", inplace = True)
df_novo["Continente"].fillna("corrigido", inplace = True)
# df_novo["Subcategoria"].fillna(df_novo["Subcategoria"].mean(), inplace=True)
```

In [27]:
```
df_novo.head(7)
```

Out[27]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continente	Unnamed: 12
0	2017-06-01 00:00:00	Home Theater System 7.1 Channel X711 Silver	Home Theater System	1109.00	367.43	Litware	1.0	1109.00	Levi	Rana	França	Europa	NaN
1	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Heidi	Patel	Estados Unidos	América do Norte	NaN
2	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Martin	Gonzalez	Chile	América do Sul	NaN
3	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Victor	Ruiz	Alemanha	Europa	NaN
4	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	5.0	1078.10	Alex	Nelson	Estados Unidos	América do Norte	NaN
5	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	1.0	215.62	Daisy	Alvarez	Estados Unidos	América do Norte	NaN
6	2017-06-01 00:00:00	180 CFM Vertical Discharge Fan X450 Black	Fans	215.62	71.44	Litware	2.0	431.24	Margaret	Wang	Estados Unidos	América do Norte	NaN

In [28]:
```
df_novo.tail(7)
```

Out[28]:

	Data da Venda	Produto	Subcategoria	PrecoUnitario	Custo Unitário	Marca	Qtd. Vendida	Faturamento	Nome Cliente	Sobrenome	País	Continent
203881	2019-08-31 00:00:00	Wireless Bluetooth Stereo Headphones M402 Red	Bluetooth Headphones	99.990000	45.980000	Northwind Traders	3.000000	299.970000	Fernando	Rodriguez	Estados Unidos	América d Nort
203882		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid
203883		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid
203884		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid
203885		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid
203886		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid
203887		corrigido	corrigido	159.804995	66.437852	corrigido	66.437852	314.754825	corrigido	corrigido	corrigido	corrigid

In [29]:
```
# Verificando quantidade de colunas e linhas
df_novo.shape
```

Out[29]:
```
(203888, 13)
```

In [30]:
```
# Verificando se ainda tem valores faltantes
df_novo.isnull().sum()
```

Out[30]:
```
Data da Venda          0
Produto                0
Subcategoria           0
PrecoUnitario          0
Custo Unitário         0
Marca                  0
Qtd. Vendida           0
Faturamento            0
Nome Cliente           0
Sobrenome              0
País                   0
Continente             0
Unnamed: 12       203886
dtype: int64
```

In [31]:
```
# Obs: Total de Linhas do Data frame
df = 203888
df_novo = 203886
subtraindo = df - df_novo
```

In [32]:
```
# Verificando
print(subtraindo)
```

```
2
```

Exploração dos dados

Analisando os dados para verificação dos seguintes Indicadores:

1 - Quantidade de Itens Vendidos;

2 - Números de Clientes Novos;

3 - Receita Total;

4 - Quantidade de Reclamações;

5 - Tempo de Vida do Cliente(LTV);

6 - Custo de Aquisição por Cliente (CAC);
7 - Cancelamento (Churn);

In [ ]:
```
df_novo
```

In [ ]:
```
# Filtrando a coluna "Data de vendas" por ano
df_novo.loc[df_novo["Data da Venda"] == 17]
```

In [ ]:
```
# 1 - Quantidade de Itens Vendidos;
tot_Itens_Vendidos = df_novo["Qtd. Vendida"].sum()
```

In [ ]:
```
# Quantidade Total de Itens Vendidos 2017, 2018, 2019
print("Total Itens Vendidos Anos 2017, 2018 2019: ", tot_Itens_Vendidos)
```