Análise de Dados de Vendas

Para se identificar padrões e tendências a fim de gerar insights e possíveis estratégias a serem colocadas em prática em uma empresa no ramo de Varejo, a análise de dados de vendas é essencial. Perguntas como "Quais categorias de produtos tem mais saída?", "Quais os produtos mais vendidos?" e "Quais são as principais características dos consumidores?" são algumas das perguntas de negócio que podem gerar bons resultados para esse tipo de análise.

Neste Jupyter Notebook trago um exemplo sucinto de análise de vendas com linguagem Python e suas principais bibliotecas para análise de dados. O dataset utilizado foi "Retail Sales Dataset. Unveiling Retail Trends: A Dive into Sales Patterns and Customer Profiles" disponível no link: https://www.kaggle.com/datasets/mohammadtalib786/retail-sales-dataset/data

In [28]:
```
# Versão Python utilizada

from platform import python_version
print("Versão Python utilizada neste Jupyter Notebook:", python_version())
```

```
Versão Python utilizada neste Jupyter Notebook: 3.9.13
```

In [29]:
```
# Importando bibliotecas necessárias

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt

%matplotlib inline
```

In [30]:
```
# Carregando o dataset "shopping_trends" em um dataframe para ser analisado

df_shop = pd.read_csv("./shopping_trends.csv")
```

In [31]:
```
# Verificando quantidade de linhas e colunas

df_shop.shape
```

Out[31]:
```
(3900, 18)
```

In [32]:
```
# Verificando amostra do dataframe

df_shop.head(10)
```

Out[32]:

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	S
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	

4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7
5	6	46	Male	Sneakers	Footwear	20	Wyoming	M	White	Summer	2.9
6	7	63	Male	Shirt	Clothing	85	Montana	M	Gray	Fall	3.2
7	8	27	Male	Shorts	Clothing	34	Louisiana	L	Charcoal	Winter	3.2
8	9	26	Male	Coat	Outerwear	97	West Virginia	L	Silver	Summer	2.6
9	10	57	Male	Handbag	Accessories	31	Missouri	M	Pink	Spring	4.8

In [33]:
```
# Verificando o tipo de dado de cada coluna

df_shop.dtypes
```

Out[33]:
```
Customer ID                int64
Age                        int64
Gender                    object
Item Purchased            object
Category                  object
Purchase Amount (USD)      int64
Location                  object
Size                      object
Color                     object
Season                    object
Review Rating            float64
Subscription Status       object
Shipping Type             object
Discount Applied          object
Promo Code Used           object
Previous Purchases         int64
Payment Method            object
Frequency of Purchases    object
dtype: object
```

In [34]:
```
# Verificando se há registros duplicados

df_shop[df_shop.duplicated()]
```

Out[34]:

Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status

In [35]:
```
# Verificando se há valores ausentes

df_shop.isnull().sum()
```

Out[35]:
```
Customer ID               0
Age                       0
Gender                    0
Item Purchased            0
Category                  0
Purchase Amount (USD)     0
Location                  0
Size                      0
Color                     0
Season                    0
Review Rating             0
Subscription Status       0
Shipping Type             0
```

```
Discount Applied          0
Promo Code Used           0
Previous Purchases        0
Payment Method            0
Frequency of Purchases    0
dtype: int64
```

In [36]:
```
# Verificando valor total de vendas

sales = df_shop["Purchase Amount (USD)"].sum()
print("O valor total de vendas foi de: U$", sales)
```

```
O valor total de vendas foi de: U$ 233081
```

In [37]:
```
# Verificando valor total de itens vendidos em cada categoria e adicionando em outro dat

df_totals = df_shop.groupby("Category")["Purchase Amount (USD)"].sum().reset_index()

# Renomeando as colunas do novo df após fazer o agrupamento

df_totals_items = df_totals.rename(columns={"Category": "Categoria", "Purchase Amount (U

# Calculando as porcentagens em relação ao valor total de vendas

df_totals_items["Porcentagem"] = round((df_totals_items["Valor Total de Vendas"] / sales
df_totals_items
```

Out[37]:

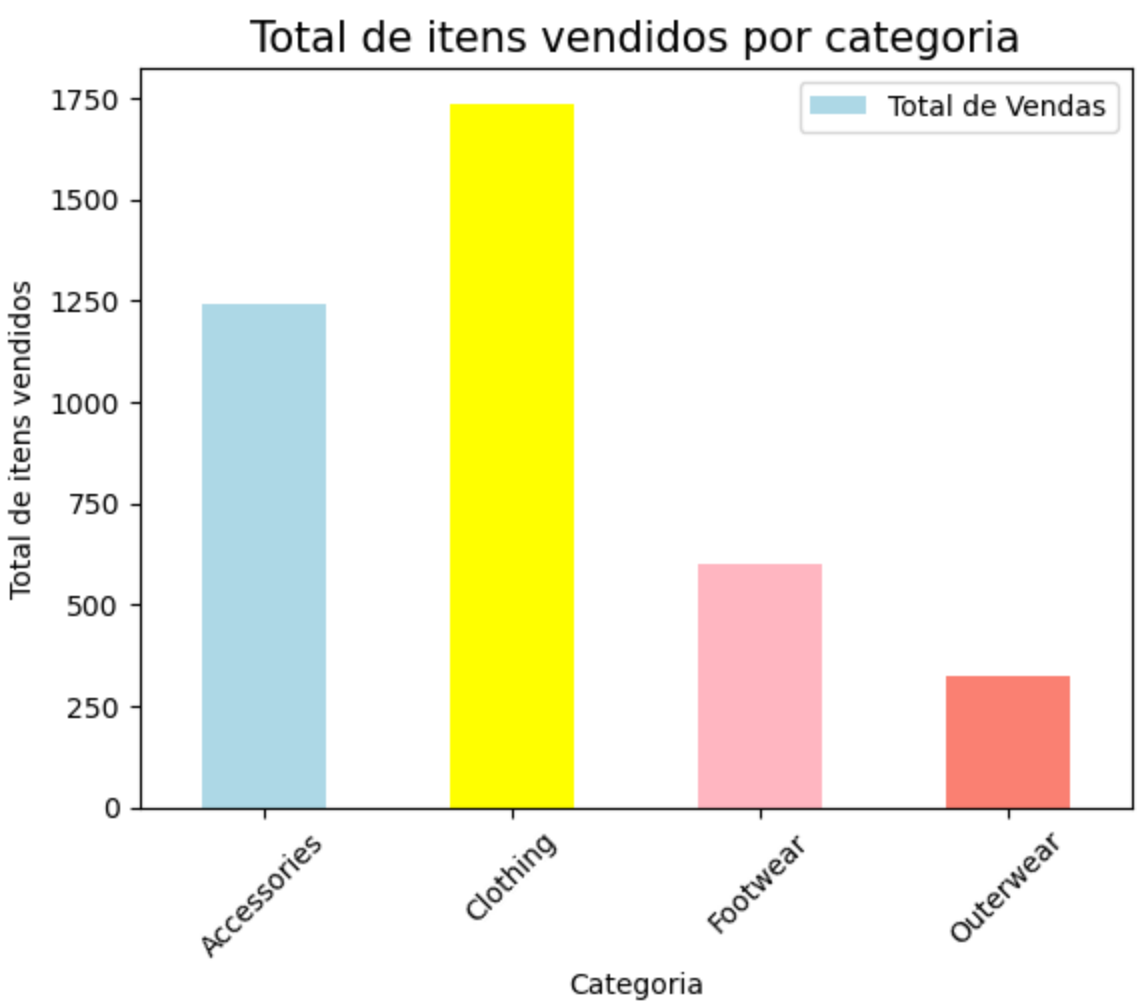
	Categoria	Valor Total de Vendas	Porcentagem
0	Accessories	74200	31.83
1	Clothing	104264	44.73
2	Footwear	36093	15.49
3	Outerwear	18524	7.95

In [38]:
```
# Verificando quantidade de itens vendidos por categoria e adicionando em outro datafram

df_cat_items = df_shop.groupby("Category")["Item Purchased"].count().reset_index()

df_cat = df_cat_items.rename(columns={"Category": "Categoria", "Item Purchased": "Total

df_cat
```

Out[38]:

	Categoria	Total de Vendas
0	Accessories	1240
1	Clothing	1737
2	Footwear	599
3	Outerwear	324

In [39]:
```
# Plotando o resultado em um gráfico de barras

df_cat.plot.bar(x = "Categoria", y = "Total de Vendas", color = ['lightblue', 'yellow',
plt.xticks(rotation = 45)
plt.xlabel("Categoria")
plt.ylabel("Total de itens vendidos")
plt.title("Total de itens vendidos por categoria", fontsize = 15)
plt.show()
```



Conclusão 1

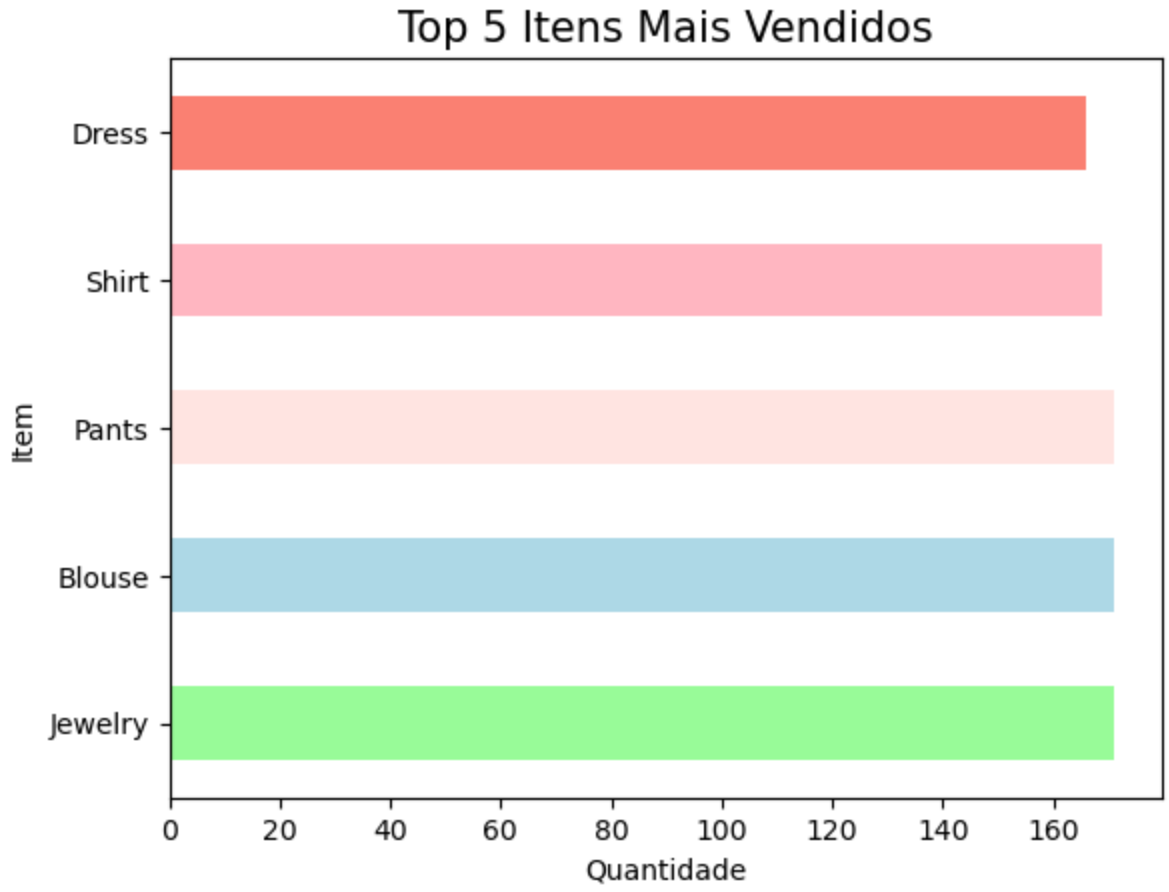
Roupas(Clothing) é a categoria mais vendida com o total de 1737 itens vendidos, representando 44,73% do valor total de vendas da empresa. Em seguida, vem as categorias de Acessórios(Accessories), Calçados(Footwear) e Agasalhos(Outerwear).

In [40]:
```
# Verificando quais itens foram os mais vendidos

df_itens1 = df_shop.groupby("Item Purchased")["Category"].count()
df_itens1.sort_values(ascending = False)
```

Out[40]:
```
Item Purchased
Jewelry       171
Blouse        171
Pants         171
Shirt         169
Dress         166
Sweater       164
Jacket        163
Coat          161
Sunglasses    161
Belt          161
Sandals       160
Socks         159
Skirt         158
Scarf         157
Shorts        157
Hat           154
Handbag       153
Hoodie        151
```

```
Shoes          150
T-shirt        147
Sneakers       145
Boots          144
Backpack       143
Gloves         140
Jeans          124
Name: Category, dtype: int64
```

In [41]:
```
# Plotando o Top 5 itens mais vendidos em um gráfico de barras

df_itens_top5 = df_itens1.sort_values(ascending = False).head(5)

df_itens_top5.plot.barh(color = ['palegreen', 'lightblue', 'mistyrose', 'lightpink', 'sa
plt.xlabel("Quantidade")
plt.ylabel("Item")
plt.title("Top 5 Itens Mais Vendidos", fontsize = 15)
plt.show()
```

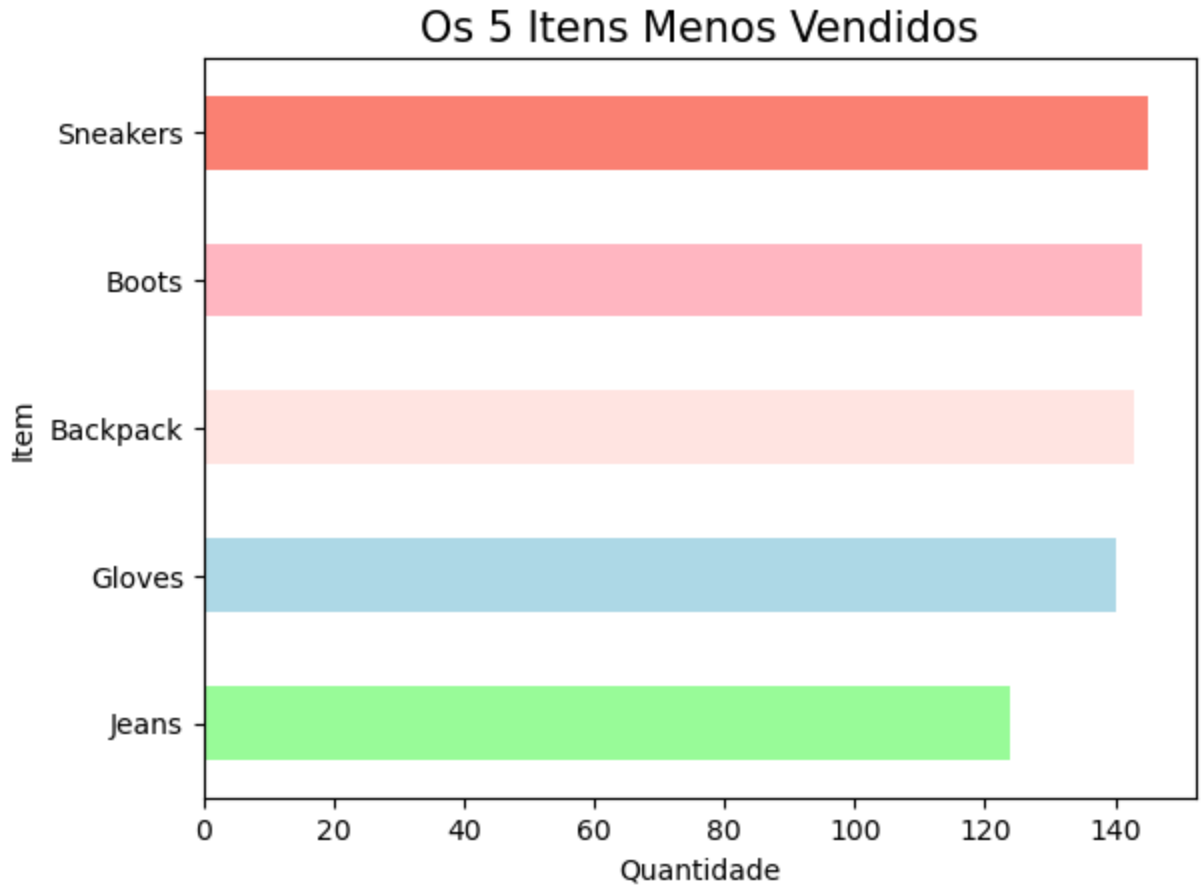


In [42]:
```
# Verificando quais itens foram os mais vendidos

df_itens2 = df_shop.groupby("Item Purchased")["Category"].count()
df_itens2.sort_values(ascending = True)
```

Out[42]:
```
Item Purchased
Jeans          124
Gloves         140
Backpack       143
Boots          144
Sneakers       145
T-shirt        147
Shoes          150
Hoodie         151
Handbag        153
Hat            154
Scarf          157
```

```
Shorts          157
Skirt           158
Socks           159
Sandals         160
Sunglasses      161
Belt            161
Coat            161
Jacket          163
Sweater         164
Dress           166
Shirt           169
Pants           171
Blouse          171
Jewelry         171
Name: Category, dtype: int64
```

In [43]:
```
# Plotando os 5 itens menos vendidos em um gráfico de barras

df_itens2_5 = df_itens2.sort_values(ascending = True).head(5)

df_itens2_5.plot.barh(color =  ['palegreen', 'lightblue', 'mistyrose', 'lightpink', 'sal
plt.xlabel("Quantidade")
plt.ylabel("Item")
plt.title("Os 5 Itens Menos Vendidos", fontsize = 15)
plt.show()
```



Conclusão 2

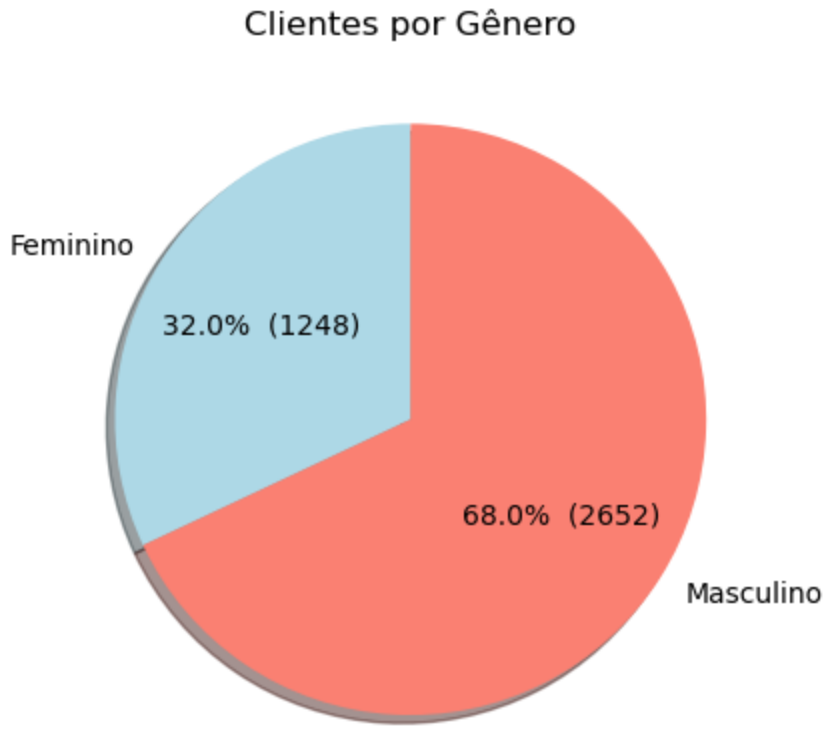
Os 5 itens mais vendidos são Vestidos(Dress), Camisetas(Shirt), Calças(Pants), Blusas(Blouse) e Jóias(Jewelry). Enquanto os 5 itens menos vendidos são Tênis(Sneakers), Botas(Boots), Mochilas(Backpack), Luvas(Gloves) e Jeans.

In [44]:
```
# Verificando gênero da maior parte dos clientes
```

```
df_gen = df_shop.groupby("Gender")["Gender"].count()
df_gen
```

Out[44]:
```
Gender
Female    1248
Male      2652
Name: Gender, dtype: int64
```

In [71]:
```
# Definindo os valores e labels do gráfico de pizza

label = ["Feminino", "Masculino"]
values = [1248, 2652]

# Definindo uma função para transformar os valores em porcentagens

def make_autopct(values):
    def my_autopct(pct):
        total = sum(values)
        val = int(round(pct*total/100.0))
        return '{p:.1f}%  ({v:d})'.format(p=pct,v=val)
    return my_autopct

# Plotando um gráfico de pizza para apresentar a porcentagem de compradores de cada gêne

plt.pie(values,
        labels = label,
        colors = ['lightblue', 'salmon'],
        autopct = make_autopct(values),
        shadow = True,
        startangle = 90)
plt.title("Clientes por Gênero")
plt.show()
```



In [46]:
```
# Verificando a idade dos clientes por gênero

df_age = df_shop[["Gender", "Age"]]
df_age
```

Out[46]:

	Gender	Age
0	Male	55
1	Male	19
2	Male	50
3	Male	21
4	Male	45
...	...	...
3895	Female	40
3896	Female	52
3897	Female	46
3898	Female	44
3899	Female	52

3900 rows × 2 columns

In [69]:
```
# Verificando dados estastísticos dos clientes masculinos

df_age_male_filter = df_age["Gender"] == "Male"
df_age_male = df_age[df_age_male_filter]
df_age_male.describe()
```

Out[69]:

	Age
count	2652.000000
mean	44.097285
std	15.328257
min	18.000000
25%	31.000000
50%	44.000000
75%	57.000000
max	70.000000

In [70]:
```
# Verificando dados estastísticos dos clientes femininos

df_age_female_filter = df_age["Gender"] == "Female"
df_age_female = df_age[df_age_female_filter]
df_age_female.describe()
```

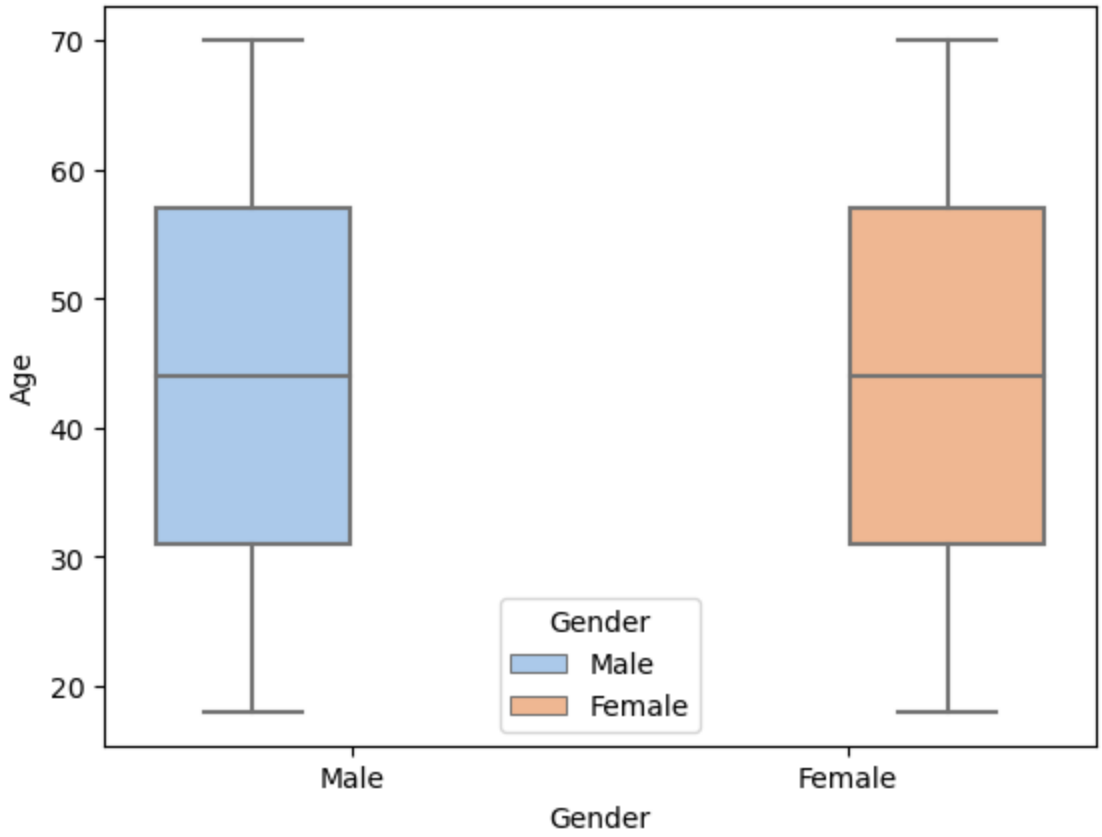
Out[70]:

	Age
count	1248.000000
mean	44.007212
std	14.953843
min	18.000000
25%	31.000000
50%	44.000000

75%	57.000000
max	70.000000

In [51]:
```
# Plotando um gráfico boxplot para mostrar as idades dos clientes em cada gênero

sns.boxplot(data = df_age, x="Gender", y="Age", hue="Gender", palette = "pastel")
```

Out[51]:
```
<AxesSubplot:xlabel='Gender', ylabel='Age'>
```



Conclusão 3

A maior parte dos clientes são do gênero masculino, representando 68% do total de clientes.

A faixa etária onde se concentram a maior parte dos clientes para ambos os gêneros está entre 31 e 57 anos.

In [74]:
```
# Verificando o valor total de vendas e a quantidade de vendas em cada estado

# Agrupando por estado o valor total e a quantidade de vendas em cada
df_st = df_shop.groupby("Location").agg({"Purchase Amount (USD)":"sum",
                                         "Customer ID": "count"}).reset_index().sort

# Renomeando as colunas do novo df após fazer o agrupamento
df_states = df_st.rename(columns={'Location': 'Estado', 'Purchase Amount (USD)': 'Valor

# Salvando apenas os 10 primeiros estados com maior valor total e quantidade de vendas
df_states_top10 = df_states.head(10)
df_states_top10
```

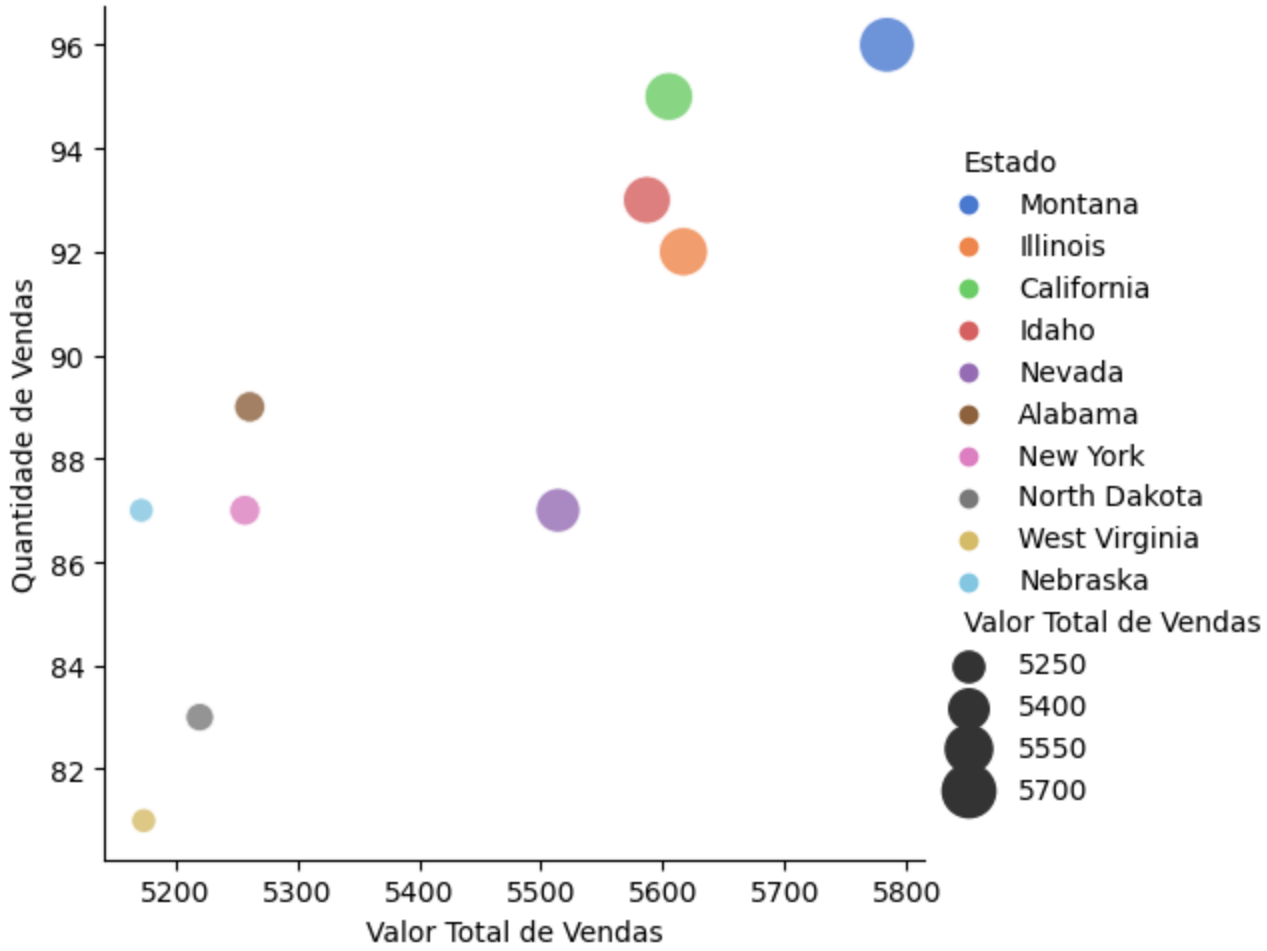
Out[74]:

	Estado	Valor Total de Vendas	Quantidade de Vendas
25	Montana	5784	96

12	Illinois	5617	92
4	California	5605	95
11	Idaho	5587	93
27	Nevada	5514	87
0	Alabama	5261	89
31	New York	5257	87
33	North Dakota	5220	83
47	West Virginia	5174	81
26	Nebraska	5172	87

In [75]:
```
# Plotando um gráfico de dispersão para mostrar os 10 primeiros estados com maior valor

sns.relplot(x="Valor Total de Vendas", y="Quantidade de Vendas", hue="Estado", size="Val
            sizes=(80, 400), alpha=.8, palette="muted",
            height=5, data=df_states_top10 )
```

Out[75]:
```
<seaborn.axisgrid.FacetGrid at 0x26dd939c820>
```



Conclusão 4

Os estados com maior número de vendas e maior valor total de vendas são, em ordem decrescente, Montana, Illinois, California, Idaho, Nevada, Alabama, New York, North Dakota, West Virginia e Nebraska.