amazon-sale-report

June 3, 2024

[1]:

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

%matplotlib inline
```

[2]:

```
df=pd.read_csv(r"C:\Users\dutta\Downloads\Amazon Sale Report.csv")
```

[3]:

```
df
```

[3]:

```
         index             Order ID      Date                        Status  \
0            0  405-8078784-5731545  04-30-22                     Cancelled
1            1  171-9198151-1101146  04-30-22  Shipped - Delivered to Buyer
2            2  404-0687676-7273146  04-30-22                       Shipped
3            3  403-9615377-8133951  04-30-22                     Cancelled
4            4  407-1069790-7240320  04-30-22                       Shipped
…        …                  …       …                           …
128971  128970  406-6001380-7673107  05-31-22                       Shipped
128972  128971  402-9551604-7544318  05-31-22                       Shipped
128973  128972  407-9547469-3152358  05-31-22                       Shipped
128974  128973  402-6184140-0545956  05-31-22                       Shipped
128975  128974  408-7436540-8728312  05-31-22                       Shipped

       Fulfilment Sales Channel ship-service-level  Category Size  \
0        Merchant     Amazon.in           Standard   T-shirt    S
1        Merchant     Amazon.in           Standard     Shirt  3XL
2          Amazon     Amazon.in          Expedited     Shirt   XL
3        Merchant     Amazon.in           Standard   Blazzer    L
4          Amazon     Amazon.in          Expedited  Trousers  3XL
…           …           …                …       …  …
128971     Amazon     Amazon.in          Expedited     Shirt   XL
128972     Amazon     Amazon.in          Expedited   T-shirt    M
128973     Amazon     Amazon.in          Expedited   Blazzer  XXL
128974     Amazon     Amazon.in          Expedited   T-shirt   XS
128975     Amazon     Amazon.in          Expedited   T-shirt    S
```

```
       Courier Status  …  currency   Amount    ship-city    ship-state  \
0          On the Way  …       INR   647.62       MUMBAI   MAHARASHTRA
1             Shipped  …       INR   406.00    BENGALURU     KARNATAKA
2             Shipped  …       INR   329.00  NAVI MUMBAI   MAHARASHTRA
3          On the Way  …       INR   753.33   PUDUCHERRY    PUDUCHERRY
4             Shipped  …       INR   574.00      CHENNAI    TAMIL NADU
…               …   …       …      …          …           …
128971        Shipped  …       INR   517.00    HYDERABAD     TELANGANA
128972        Shipped  …       INR   999.00     GURUGRAM       HARYANA
128973        Shipped  …       INR   690.00    HYDERABAD     TELANGANA
128974        Shipped  …       INR  1199.00        Halol       Gujarat
128975        Shipped  …       INR   696.00       Raipur  CHHATTISGARH

       ship-postal-code  ship-country    B2B  fulfilled-by New  PendingS
0              400081.0            IN  False     Easy Ship NaN       NaN
1              560085.0            IN  False     Easy Ship NaN       NaN
2              410210.0            IN   True           NaN NaN       NaN
3              605008.0            IN  False     Easy Ship NaN       NaN
4              600073.0            IN  False           NaN NaN       NaN
…                 …           …    …           …  . .      …
128971         500013.0            IN  False           NaN NaN       NaN
128972         122004.0            IN  False           NaN NaN       NaN
128973         500049.0            IN  False           NaN NaN       NaN
128974         389350.0            IN  False           NaN NaN       NaN
128975         492014.0            IN  False           NaN NaN       NaN

[128976 rows x 21 columns]
```

```
[4]: df=df.drop(columns=["index"])
```

```
[5]: df.head()
```

```
[5]:
              Order ID      Date                        Status Fulfilment  \
0  405-8078784-5731545  04-30-22                     Cancelled   Merchant
1  171-9198151-1101146  04-30-22  Shipped - Delivered to Buyer   Merchant
2  404-0687676-7273146  04-30-22                       Shipped     Amazon
3  403-9615377-8133951  04-30-22                     Cancelled   Merchant
4  407-1069790-7240320  04-30-22                       Shipped     Amazon

  Sales Channel ship-service-level  Category Size Courier Status  Qty   \
0     Amazon.in           Standard   T-shirt    S     On the Way    0
1     Amazon.in           Standard     Shirt  3XL        Shipped    1
2     Amazon.in          Expedited     Shirt   XL        Shipped    1
3     Amazon.in           Standard   Blazzer    L     On the Way    0
4     Amazon.in          Expedited  Trousers  3XL        Shipped    1

  currency  Amount    ship-city   ship-state  ship-postal-code ship-country   \
0      INR  647.62       MUMBAI  MAHARASHTRA           400081.0            IN
1      INR  406.00    BENGALURU    KARNATAKA           560085.0            IN
2      INR  329.00  NAVI MUMBAI  MAHARASHTRA           410210.0            IN
3      INR  753.33   PUDUCHERRY   PUDUCHERRY           605008.0            IN
4      INR  574.00      CHENNAI   TAMIL NADU           600073.0            IN

     B2B fulfilled-by  New  PendingS
0  False    Easy Ship  NaN       NaN
1  False    Easy Ship  NaN       NaN
2   True          NaN  NaN       NaN
3  False    Easy Ship  NaN       NaN
4  False          NaN  NaN       NaN
```

[6]:

```
df.isnull().sum()
```

[6]:

```
Order ID                   0
Date                       0
Status                     0
Fulfilment                 0
Sales Channel              0
ship-service-level         0
Category                   0
Size                       0
Courier Status             0
Qty                        0
currency                7800
Amount                  7800
ship-city                 35
ship-state                35
ship-postal-code          35
ship-country              35
B2B                        0
fulfilled-by           89713
New                   128976
PendingS              128976
dtype: int64
```

[7]:

```
df.duplicated()
```

[7]:

```
0         False
1         False
2         False
3         False
4         False
          …
128971    False
128972    False
```

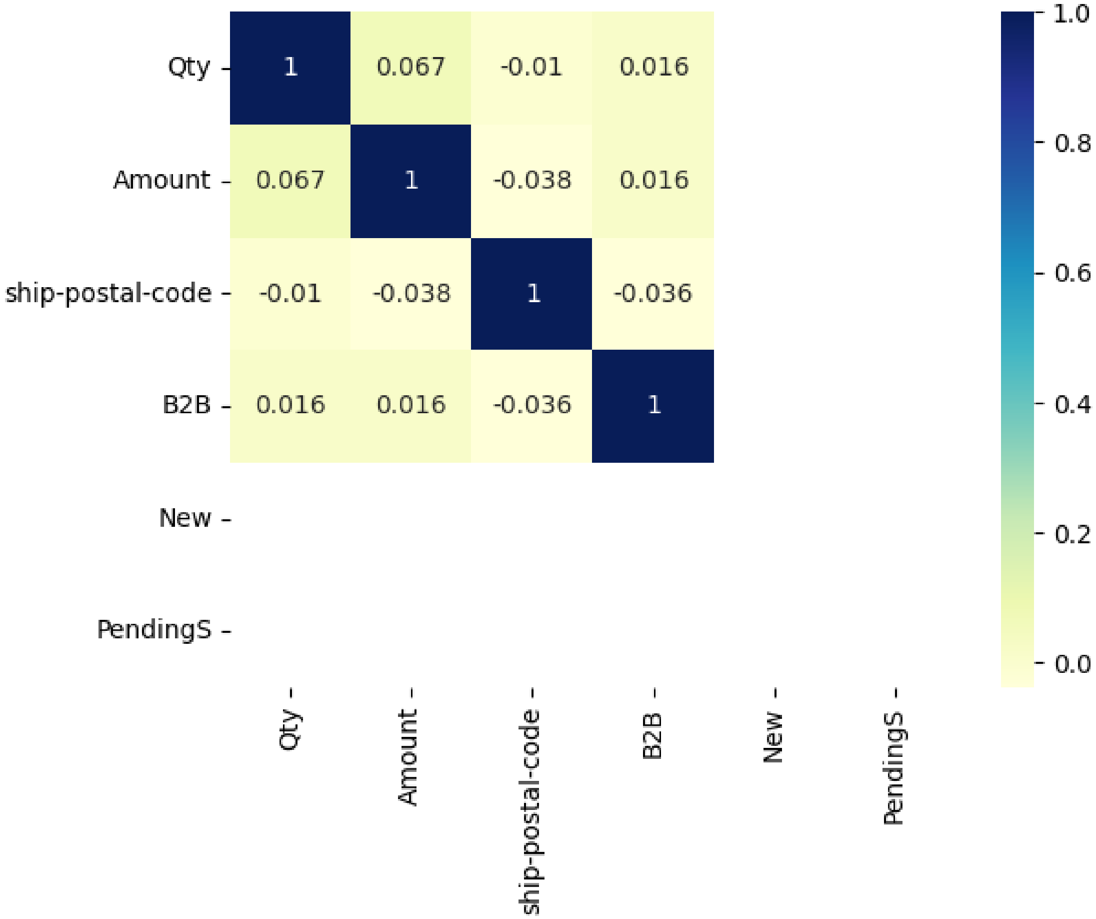
```
128973    False
128974    False
128975    False
Length: 128976, dtype: bool
```

[8]:
```
df.corr(numeric_only=True)
```

[8]:
```
                        Qty    Amount  ship-postal-code       B2B  New  \
Qty                1.000000  0.066750         -0.010231  0.015810  NaN
Amount             0.066750  1.000000         -0.038423  0.015560  NaN
ship-postal-code  -0.010231 -0.038423          1.000000 -0.035792  NaN
B2B                0.015810  0.015560         -0.035792  1.000000  NaN
New                     NaN       NaN               NaN       NaN  NaN
PendingS                NaN       NaN               NaN       NaN  NaN

                  PendingS
Qty                    NaN
Amount                 NaN
ship-postal-code       NaN
B2B                    NaN
New                    NaN
PendingS               NaN
```

[9]:
```
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap="YlGnBu")
```

[9]:
```
<Axes: >
```



[10]:

```
#converting 'Date' column to datetime fromat

df['Date'] = pd.to_datetime(df['Date'], format='%m-%d-%y', errors ='coerce')
```

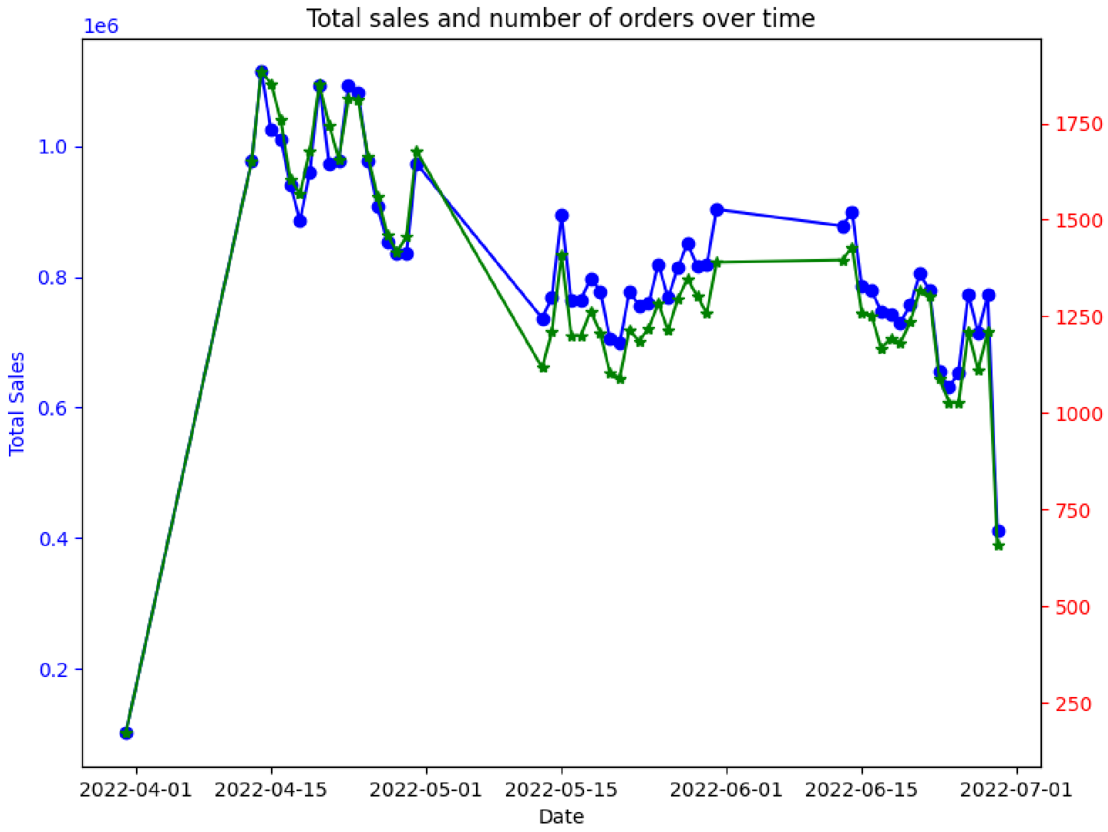
[11]:

```
#Aggregate total sales and number of orders by date

sales_overview = df.groupby('Date').agg({'Amount':'sum','Order ID':'count'}).
  ↪reset_index()
sales_overview.rename(columns={'Order ID':'Number of Orders'}, inplace= True)
```

[12]:

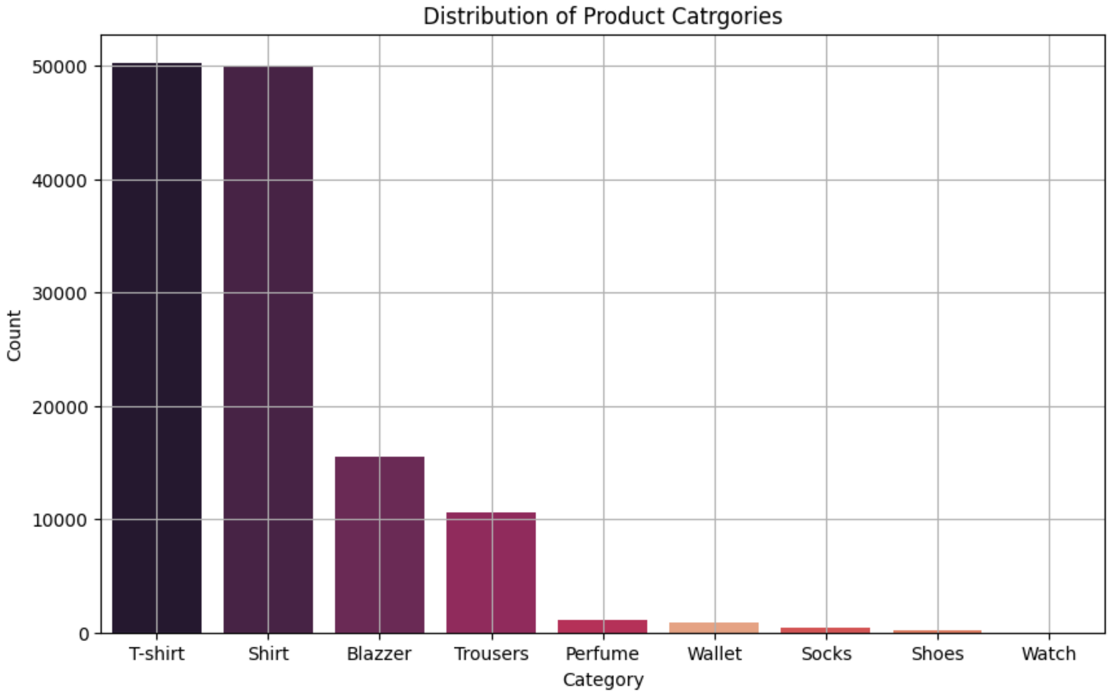
```
#plotting that sales and number of orders

fig, ax1 = plt.subplots(figsize=(8,6))

ax1.plot(sales_overview['Date'], sales_overview['Amount'],color='b',marker='o',␣
  ↪label='Total Sales')
ax1.set_xlabel('Date')
```

```
ax1.set_ylabel('Total Sales', color='b')
ax1.tick_params('y',colors='b')

ax2= ax1.twinx()
ax2.plot(sales_overview['Date'],sales_overview['Number of␣
 ↪Orders'],color='g',marker='*',label='Number of Orders')
ax2.tick_params('y',colors='r')

fig.tight_layout()
plt.title('Total sales and number of orders over time')
plt.show()
```

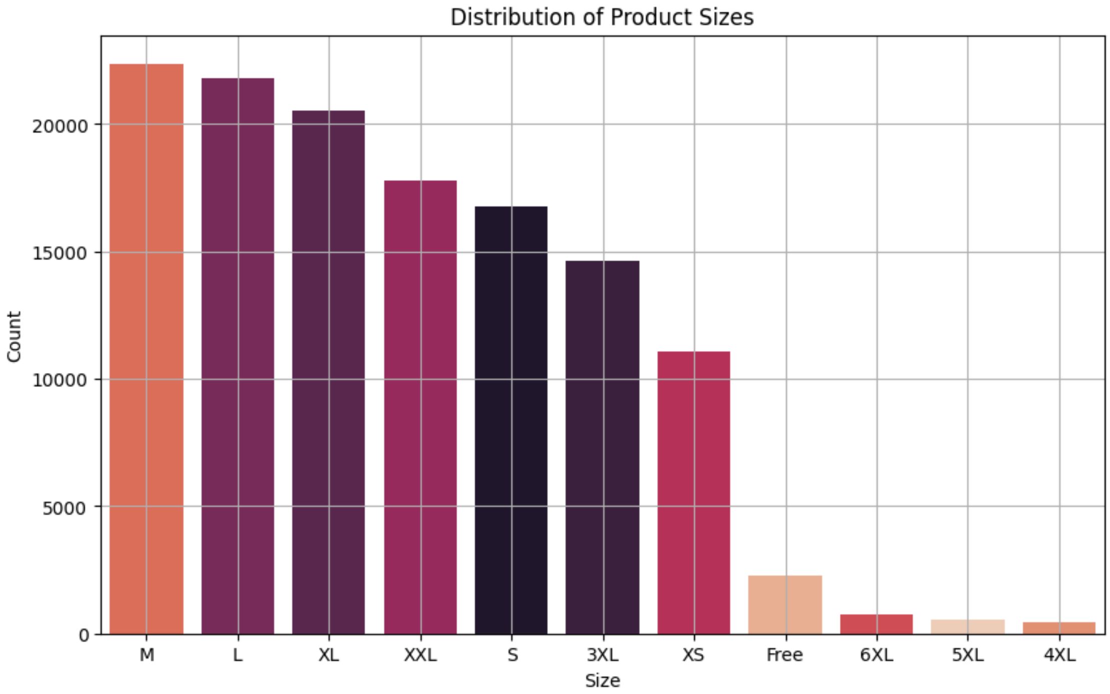


[41]:

```
#plotting distribution of product categories
unique_categories = df['Category'].nunique()
custom_palette = sns.color_palette("rocket", unique_categories)
plt.figure(figsize=(10,6))
sns.countplot(data=df, x='Category', hue='Category', order=df['Category'].
 ↪value_counts().index, palette=custom_palette, legend=False)

plt.title('Distribution of Product Catrgories')
plt.xlabel('Category')
plt.ylabel('Count')
plt.grid(True)
plt.show()
```

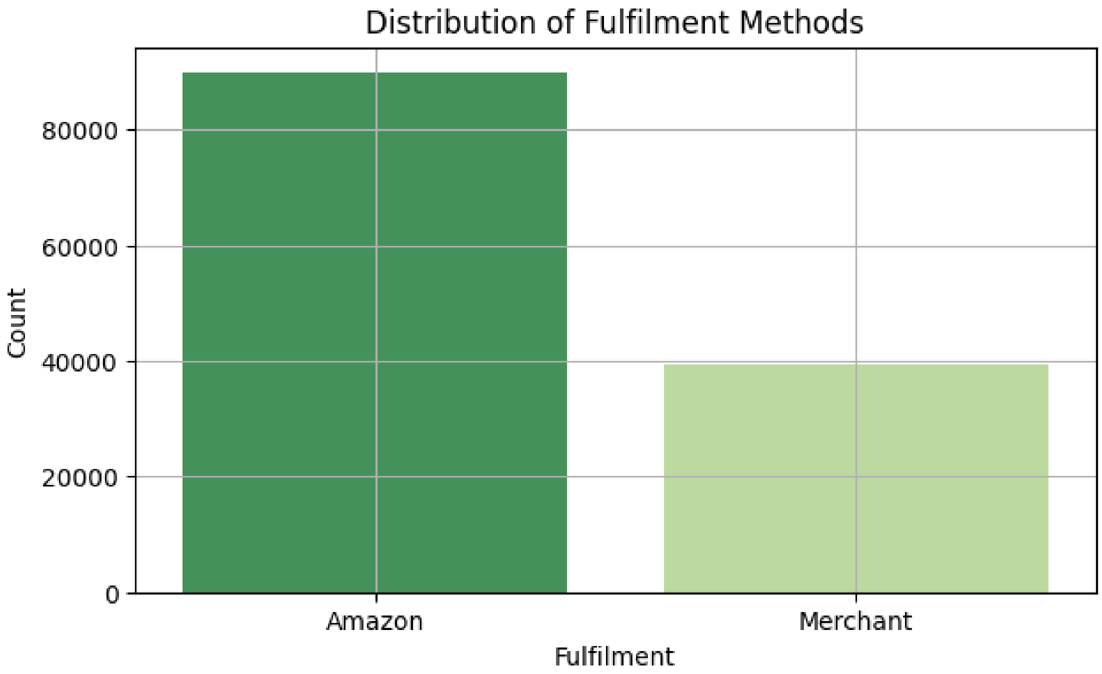


[61]:
```
#plotting distribution of product size
unique_sizes = df['Size'].nunique()
custom_palette = sns.color_palette("rocket", unique_sizes)
plt.figure(figsize=(10,6))
sns.countplot(data=df, x='Size', hue='Size', order=df['Size'].value_counts().
 ↪index, palette=custom_palette, legend=False)

plt.title('Distribution of Product Sizes')
plt.xlabel('Size')
plt.ylabel('Count')
plt.grid(True)
plt.show()
```



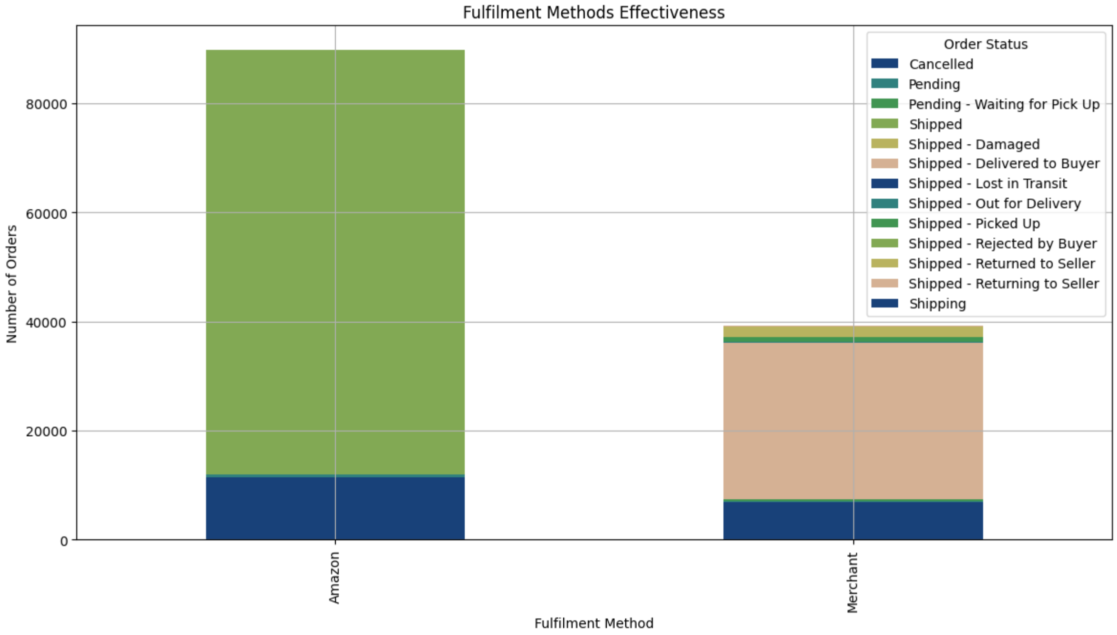
[51]:

```
#plotting distribution of fulfillment methods
unique_fulfillment = df['Fulfilment'].nunique()
custom_palette = sns.color_palette("YlGn", unique_fulfillment)
plt.figure(figsize=(7,4))
sns.countplot(data=df, x='Fulfilment', hue='Fulfilment', order=df['Fulfilment'].
 ↪value_counts().index, palette=custom_palette, legend=False)

plt.title('Distribution of Fulfilment Methods')
plt.xlabel('Fulfilment')
plt.ylabel('Count')
plt.grid(True)
plt.show()
```

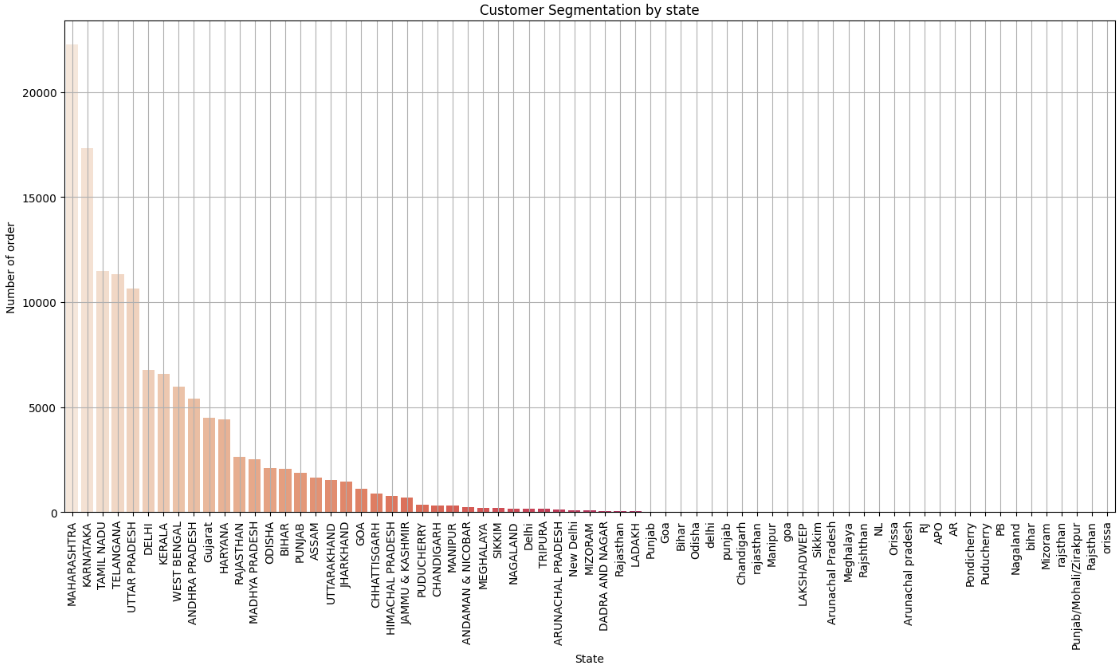


[16]:
```
#Analyzing the effectiveness based on "Status"

custom_palette = sns.color_palette("gist_earth",6)
plt.figure(figsize=(6,4))
fulfillment_effectiveness = df.groupby(['Fulfilment','Status']).size().
 ↪unstack().fillna(0)
fulfillment_effectiveness.plot(kind='bar',␣
 ↪stacked=True,figsize=(14,7),color=custom_palette)

plt.title('Fulfilment Methods Effectiveness')
plt.xlabel('Fulfilment Method')
plt.ylabel('Number of Orders')
plt.grid(True)
plt.legend(title='Order Status')
plt.show()
```

```
<Figure size 600x400 with 0 Axes>
```

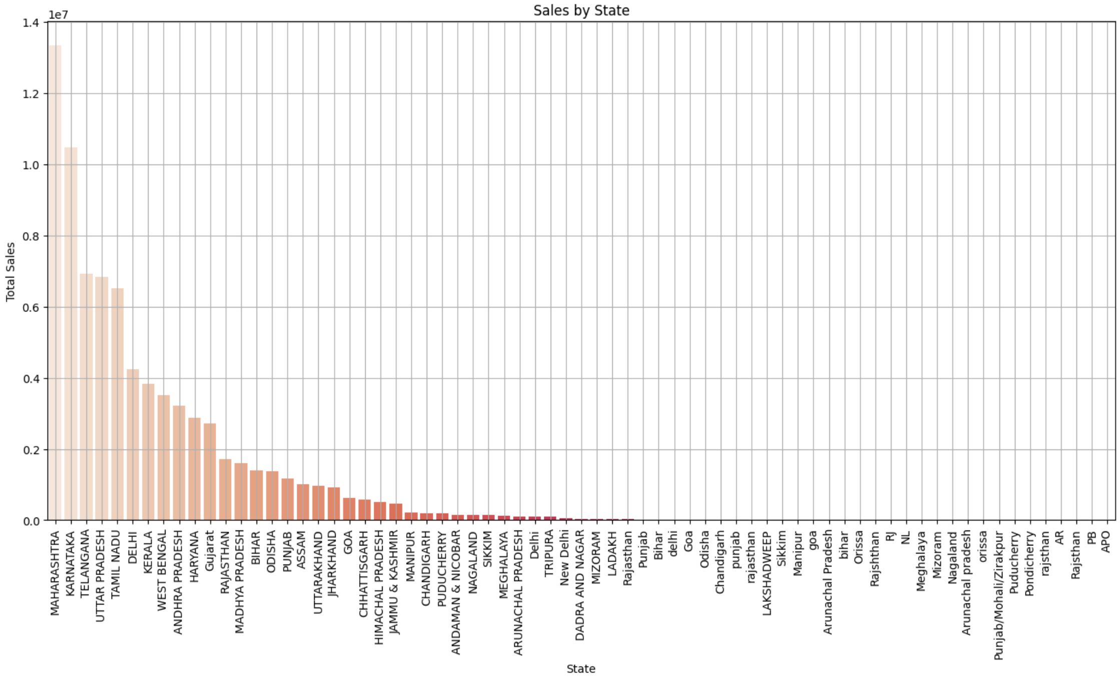


[57]:
```
#Segmentation by state

state_segmentation = df['ship-state'].value_counts().reset_index()
state_segmentation.columns = ['State','Number of Orders']
unique_states = state_segmentation['State'].nunique()
custom_palette = sns.color_palette("rocket_r", unique_states)
plt.figure(figsize=(17,8))
sns.barplot(data=state_segmentation, x='State', y='Number of Orders',␣
  ↪hue='State', palette=custom_palette, legend=False)
plt.title('Customer Segmentation by state')
plt.xlabel('State')
plt.ylabel('Number of order')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()
```

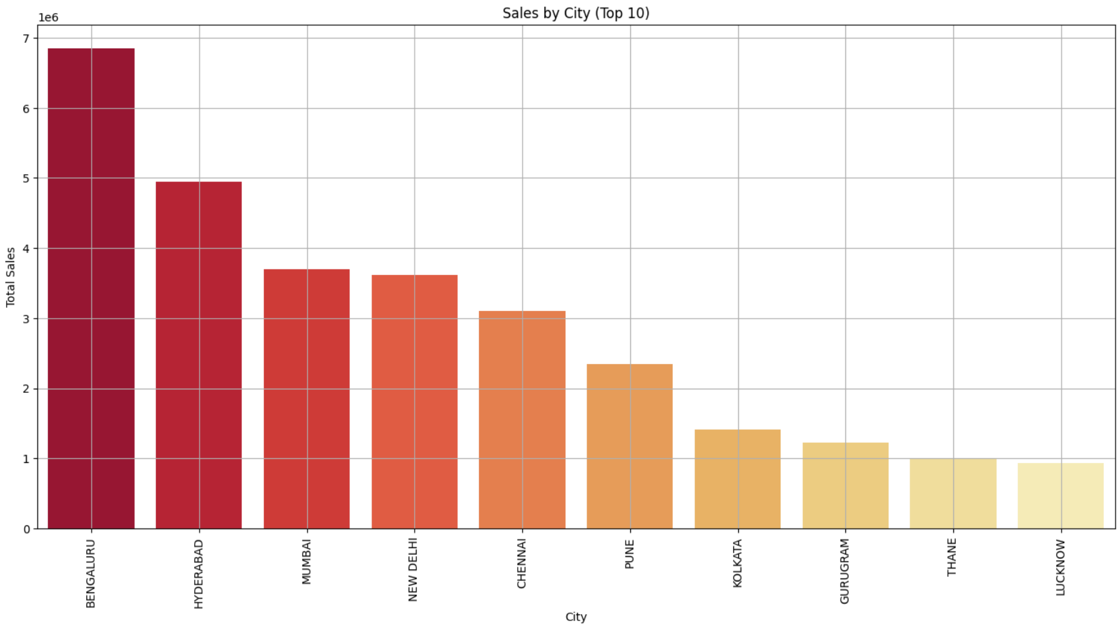


[18]:
```
#Aggregate sales by state and city

geo_sales = df.groupby(['ship-state', 'ship-city']).agg({'Amount':'sum'}).
 ↪reset_index()

plt.show()
```

[66]:
```
#plotting sales by state



state_sales = geo_sales.groupby('ship-state').agg({'Amount':'sum'}).
 ↪reset_index()

state_sales = state_sales.sort_values('Amount', ascending= False)
unique_states = state_sales['ship-state'].nunique()
custom_palette = sns.color_palette("rocket_r",unique_states)
plt.figure(figsize=(17,8))
sns.barplot(data=state_sales, x='ship-state', y='Amount', hue='ship-state',␣
 ↪palette=custom_palette, legend=False)
plt.title('Sales by State')
plt.xlabel('State')
plt.ylabel('Total Sales')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()
```



[63]:
```python
#plotting sales by city(top 10 cities)


state_sales = geo_sales.groupby('ship-city').agg({'Amount':'sum'}).reset_index()

state_sales = state_sales.sort_values('Amount', ascending= False).head(10)
unique_cities = state_sales['ship-city'].nunique()
custom_palette = sns.color_palette("YlOrRd_", unique_cities)
plt.figure(figsize=(17,8))
sns.barplot(data=state_sales, x='ship-city', y='Amount', hue='ship-city',␣
  ↪palette=custom_palette, legend=False)
plt.title('Sales by City (Top 10)')
plt.xlabel('City')
plt.ylabel('Total Sales')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()
```


Sales by City (Top 10)
1e6
7
6
5
4
3
2
1
0
Total Sales
BENGALURU
HYDERABAD
MUMBAI
NEW DELHI
CHENNAI
PUNE
KOLKATA
GURUGRAM
THANE
LUCKNOW
City

[ ]: