In [1]:

```python
import numpy as np
import pandas as pd
import ast
import plotly.express as px
from plotly import graph_objects as go
```

In [2]:

```python
df = pd.read_csv("flipkart_com-ecommerce_sample.csv")
df
```

Out[2]:

	uniq_id	crawl_timestamp	product_url	product_name	product_category_tree	pid	retail_price	discounted_price	image
0	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["http://img5a.flixcart.com/image/short/u/4/a/...
1	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000	http://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["http://img6a.flixcart.com/image/sofa-bed/j/f...
2	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000	http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["http://img5a.flixcart.com/image/shoe/7/z/z/r...
3	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["http://img5a.flixcart.com/image/short/6/2/h/...
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000	http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	["http://img5a.flixcart.com/image/pet-shampoo/...
...	...	...	...	...	...	...	...	...	...
19997	93e9d343837400ce0d7980874ece471c	2015-12-01 10:15:43 +0000	http://www.flipkart.com/elite-collection-mediu...	Elite Collection Medium Acrylic Sticker	["Baby Care >> Baby & Kids Gifts >> Stickers >...	STIE7VAYDKQZEBSD	1299.0	999.0	["http://img5a.flixcart.com/image/sticker/b/s/...
19998	669e79b8fa5d9ae020841c0c97d5e935	2015-12-01 10:15:43 +0000	http://www.flipkart.com/elite-collection-mediu...	Elite Collection Medium Acrylic Sticker	["Baby Care >> Baby & Kids Gifts >> Stickers >...	STIE8YSVEPPCZ42Y	1499.0	1199.0	["http://img5a.flixcart.com/image/sticker/4/2/...
19999	cb4fa87a874f715fff567f7b7b3be79c	2015-12-01 10:15:43 +0000	http://www.flipkart.com/elite-collection-mediu...	Elite Collection Medium Acrylic Sticker	["Baby Care >> Baby & Kids Gifts >> Stickers >...	STIE88KN9ZDSGZKY	1499.0	999.0	["http://img6a.flixcart.com/image/sticker/z/k/...
20000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20001	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

20002 rows × 15 columns

In [3]:

```python
df.head()
```

Out[3]:

	uniq_id	crawl_timestamp	product_url	product_name	product_category_tree	pid	retail_price	discounted_price	image	is_
0	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["http://img5a.flixcart.com/image/short/u/4/a/...	
1	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000	http://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["http://img6a.flixcart.com/image/sofa-bed/j/f...	
2	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000	http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["http://img5a.flixcart.com/image/shoe/7/z/z/r...	
3	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["http://img5a.flixcart.com/image/short/6/2/h/...	
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000	http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	["http://img5a.flixcart.com/image/pet-shampoo/...	

In [4]:

```python
df.shape
```

Out[4]:

```
(20002, 15)
```

In [5]:

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20002 entries, 0 to 20001
Data columns (total 15 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   uniq_id                  20000 non-null  object
 1   crawl_timestamp          20000 non-null  object
 2   product_url              20000 non-null  object
 3   product_name             20000 non-null  object
 4   product_category_tree    20000 non-null  object
 5   pid                      20000 non-null  object
 6   retail_price             19922 non-null  float64
 7   discounted_price         19922 non-null  float64
 8   image                    19997 non-null  object
 9   is_FK_Advantage_product  20000 non-null  object
 10  description              19998 non-null  object
 11  product_rating           20000 non-null  object
 12  overall_rating           20000 non-null  object
 13  brand                    14136 non-null  object
 14  product_specifications   19986 non-null  object
dtypes: float64(2), object(13)
memory usage: 2.3+ MB
```

In [6]:

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

Out[6]:

```
uniq_id                       2
crawl_timestamp               2
product_url                   2
product_name                  2
product_category_tree         2
pid                           2
retail_price                 80
discounted_price             80
image                         5
is_FK_Advantage_product       2
description                   4
product_rating                2
overall_rating                2
brand                      5866
product_specifications       16
dtype: int64
```

In [7]:

```python
df["retail_price"].fillna(df["retail_price"].median(),inplace=True)
df["discounted_price"].fillna(df["discounted_price"].median(),inplace=True)
df.head()
```

Out[7]:

	uniq_id	crawl_timestamp	product_url	product_name	product_category_tree	pid	retail_price	discounted_price	image	is_
0	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["http://img5a.flixcart.com/image/short/u/4/a/...	
1	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000	http://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["http://img6a.flixcart.com/image/sofa-bed/j/f...	
2	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000	http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["http://img5a.flixcart.com/image/shoe/7/z/z/r...	
3	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["http://img5a.flixcart.com/image/short/6/2/h/...	
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000	http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	["http://img5a.flixcart.com/image/pet-shampoo/...	

In [8]:

```python
x=df['retail_price']-df['discounted_price']
y=(x/df['retail_price'])*100
df['discount_percentage']=y
```

In [9]:

```python
df['timestamp'] = pd.to_datetime(df['crawl_timestamp'], errors='coerce')

# Extract the time part of the timestamp (handling NaT gracefully)
df['Time'] = df['timestamp'].apply(lambda x: x.time() if pd.notnull(x) else None)

# Extract the date part of the timestamp (handling NaT gracefully)
df['date'] = df['timestamp'].apply(lambda x: x.date() if pd.notnull(x) else None)


df.head()
```

Out[9]:

	uniq_id	crawl_timestamp	product_url	product_name	product_category_tree	pid	retail_price	discounted_price	image	is_
0	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["http://img5a.flixcart.com/image/short/u/4/a/...	
1	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000	http://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["http://img6a.flixcart.com/image/sofa-bed/j/f...	
2	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000	http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["http://img5a.flixcart.com/image/shoe/7/z/z/r...	
3	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["http://img5a.flixcart.com/image/short/6/2/h/...	
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000	http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	["http://img5a.flixcart.com/image/pet-shampoo/...	

In [10]:

```python
# Extract the main category from 'product_category_tree', with error handling
df['main_category'] = df['product_category_tree'].apply(lambda x: x.split('>>')[0][2:] if isinstance(x, str) else None)

df.head()
```

Out[10]:

	uniq_id	crawl_timestamp	product_url	product_name	product_category_tree	pid	retail_price	discounted_price	image	is_
0	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2FF9KEDEFGF	999.0	379.0	["http://img5a.flixcart.com/image/short/u/4/a/...	
1	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000	http://www.flipkart.com/fabhomedecor-fabric-do...	FabHomeDecor Fabric Double Sofa Bed	["Furniture >> Living Room Furniture >> Sofa B...	SBEEH3QGU7MFYJFY	32157.0	22646.0	["http://img6a.flixcart.com/image/sofa-bed/j/f...	
2	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000	http://www.flipkart.com/aw-bellies/p/itmeh4grg...	AW Bellies	["Footwear >> Women's Footwear >> Ballerinas >...	SHOEH4GRSUBJGZXE	999.0	499.0	["http://img5a.flixcart.com/image/shoe/7/z/z/r...	
3	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000	http://www.flipkart.com/alisha-solid-women-s-c...	Alisha Solid Women's Cycling Shorts	["Clothing >> Women's Clothing >> Lingerie, Sl...	SRTEH2F6HUZMQ6SJ	699.0	267.0	["http://img5a.flixcart.com/image/short/6/2/h/...	
4	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000	http://www.flipkart.com/sicons-all-purpose-arn...	Sicons All Purpose Arnica Dog Shampoo	["Pet Supplies >> Grooming >> Skin & Coat Care...	PSOEH3ZYDMSYARJ5	220.0	210.0	["http://img5a.flixcart.com/image/pet-shampoo/...	

Counting top 10 products in main category column

In [11]:

```python
df['main_category']
```

Out[11]:

```
0            Clothing
1           Furniture
2            Footwear
3            Clothing
4        Pet Supplies
             ...
19997       Baby Care
19998       Baby Care
19999       Baby Care
20000            None
20001            None
Name: main_category, Length: 20002, dtype: object
```

In [12]:

```python
n = 10
top_products = pd.DataFrame(df['main_category'].value_counts()[:n]).reset_index()
print("Before renaming:")
print(top_products)
```

```
Before renaming:
                main_category  count
0                    Clothing   6198
1                   Jewellery   3531
2                    Footwear   1227
3        Mobiles & Accessories   1099
4                  Automotive   1012
5   Home Decor & Festive Needs    929
6      Beauty and Personal Care    710
7              Home Furnishing    700
8             Kitchen & Dining    647
9                    Computers    578
```

In [13]:

```python
# Rename columns
top_products.columns = ['Top_Products', 'Total_count']
print("\nAfter renaming:")
print(top_products)
```

```
After renaming:
                 Top_Products  Total_count
0                    Clothing         6198
1                   Jewellery         3531
2                    Footwear         1227
3        Mobiles & Accessories         1099
4                  Automotive         1012
5   Home Decor & Festive Needs          929
6      Beauty and Personal Care          710
7              Home Furnishing          700
8             Kitchen & Dining          647
9                    Computers          578
```

Top 10 main brands being purchased

In [14]:

```python
df['brand']
```

Out[14]:

```
0                  Alisha
1            FabHomeDecor
2                      AW
3                  Alisha
4                  Sicons
              ...
19997    Elite Collection
19998    Elite Collection
19999    Elite Collection
20000                 NaN
20001                 NaN
Name: brand, Length: 20002, dtype: object
```

In [15]:

```python
#Top 10 main brands being purchased

n = 10
top_brands=pd.DataFrame(df['brand'].value_counts()[:n]).reset_index()
print("Before renaming:")
print(top_brands)
```

```
Before renaming:
          brand  count
0   Allure Auto    469
1       Regular    313
2        Voylla    299
3          Slim    288
4  TheLostPuppy    229
5    Karatcraft    211
6         Black    167
7         White    155
8  DailyObjects    144
9      Speedwav    141
```

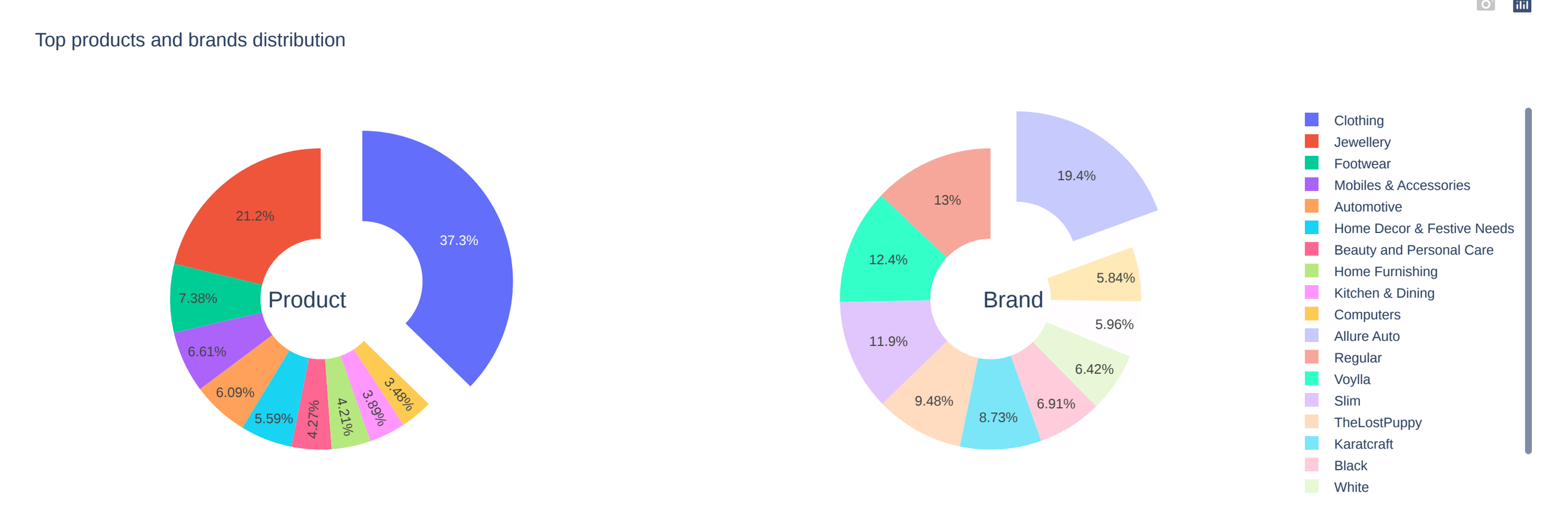
In [16]:

```python
# Rename columns
top_brands.columns = ['Top Brands', 'Total count']
print("\nAfter renaming:")
print(top_brands)
```

```
After renaming:
     Top Brands  Total count
0   Allure Auto          469
1       Regular          313
2        Voylla          299
3          Slim          288
4  TheLostPuppy          229
5    Karatcraft          211
6         Black          167
7         White          155
8  DailyObjects          144
9      Speedwav          141
```

In [17]:

```python
from plotly.subplots import make_subplots


label1=top_products['Top_Products']

value1=top_products['Total_count']

label2=top_brands['Top Brands']

value2=top_brands['Total count']


#Create subplots

fig_both = make_subplots(rows=1, cols=2, specs=[[{'type': 'domain'}, {'type': 'domain'}]])

fig_both.add_trace(go.Pie(labels=label1, values=value1, name="Top Products", pull=[0.3, 0, 0, 0]),1,1)

fig_both.add_trace(go.Pie(labels=label2, values=value2, name="Total Brands", pull=[0.3, 0, 0, 0]), 1, 2)


#use hole to create a donut-like pie chart

fig_both.update_traces(hole=.4, hoverinfo="label+percent+name")

#fig_both.update_traces (hoverinfo="Label+percent+name")

fig_both.update_layout(

title_text="Top products and brands distribution",
    #Add annotations in the center of the donut pies
    annotations=[dict(text='Product', x=0.18, y=0.5, font_size=20, showarrow=False),
                 dict(text='Brand', x=0.82, y=0.5, font_size=20, showarrow=False)])
```



High Discount Brand Analysis

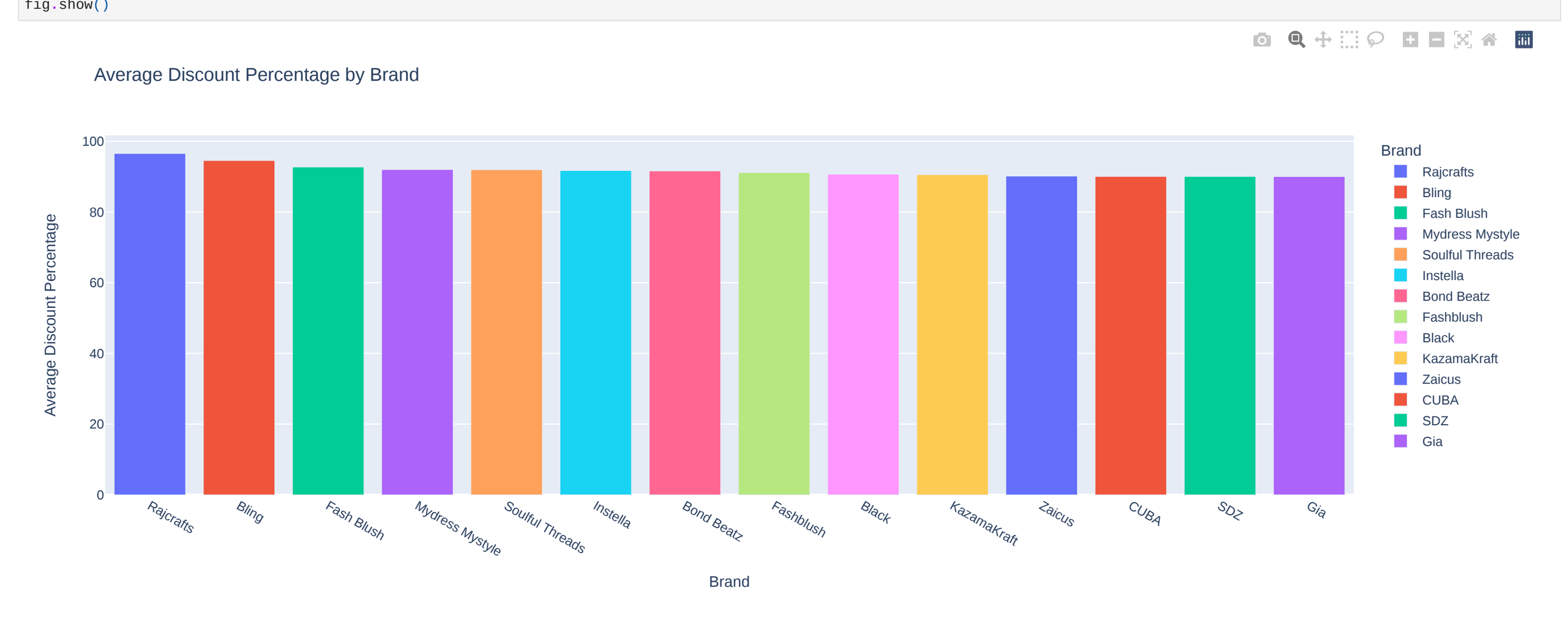
In [18]:

```python
# Filter for high discounts (discount_percentage > 90)
df_discount = df.query('discount_percentage > 90')

# Drop rows with missing values
df_discount = df_discount.dropna()

# Correct spelling errors in the 'brand' column
df_discount["brand"].replace('FashBlush', 'Fash Blush', inplace=True)

# Calculate average discount percentage by brand
max_discount = (df_discount
                .groupby('brand')[['discount_percentage']]
                .mean()
                .sort_values(by='discount_percentage', ascending=False)
                .reset_index())

print(max_discount)
```

```
              brand  discount_percentage
0         Rajcrafts            96.533333
1             Bling            94.548458
2        Fash Blush            92.711714
3   Mydress Mystyle            91.991992
4   Soulful Threads            91.952663
5          Instella            91.719745
6        Bond Beatz            91.596639
7         Fashblush            91.132525
8             Black            90.681676
9       KazamaKraft            90.565618
10           Zaicus            90.143281
11             CUBA            90.045023
12              SDZ            90.045023
13              Gia            90.020004
```

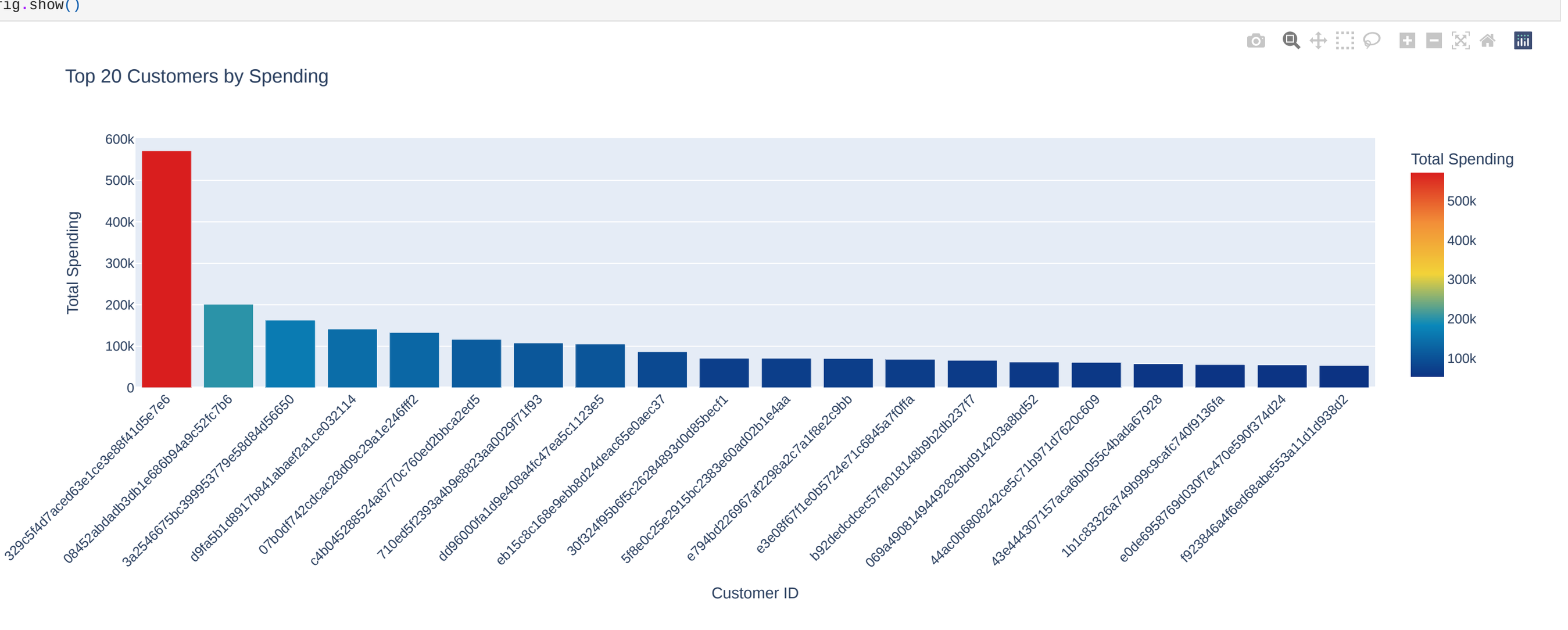
In [19]:

```python
# Create a bar plot with enhancements
fig = px.bar(
    max_discount,
    x='brand',
    y='discount_percentage',
    color='brand',
    color_discrete_sequence=px.colors.qualitative.Plotly,
    title='Average Discount Percentage by Brand',
    labels={'discount_percentage': 'Average Discount Percentage', 'brand': 'Brand'}
)

# Update layout for better readability
fig.update_layout(
    xaxis_title='Brand',
    yaxis_title='Average Discount Percentage'
)

# Show the plot
fig.show()
```



In [20]:

```python
# Group by customer and calculate total spending
df_customer = df.groupby("uniq_id")[["discounted_price"]].sum().sort_values(by=['discounted_price'], ascending=[False])

# Select the top 20 customers with the highest spending
top_20_customers = df_customer.head(20)

# Ensure 'uniq_id' is a column in the DataFrame
top_20_customers = top_20_customers.reset_index()  # Reset index to make 'uniq_id' a column

# Create a bar plot
fig = px.bar(
    top_20_customers,
    x='uniq_id',
    y='discounted_price',
    color='discounted_price',
    color_continuous_scale=px.colors.diverging.Portland,  # Customize the color scale if needed
    title='Top 20 Customers by Spending',
    labels={'discounted_price': 'Total Spending', 'uniq_id': 'Customer ID'}
)

# Update layout for better readability
fig.update_layout(
    xaxis_title='Customer ID',
    yaxis_title='Total Spending',
    xaxis_tickangle=-45  # Optional: Rotates x-axis labels for better readability
)

# Show the plot
fig.show()
```



In [21]:

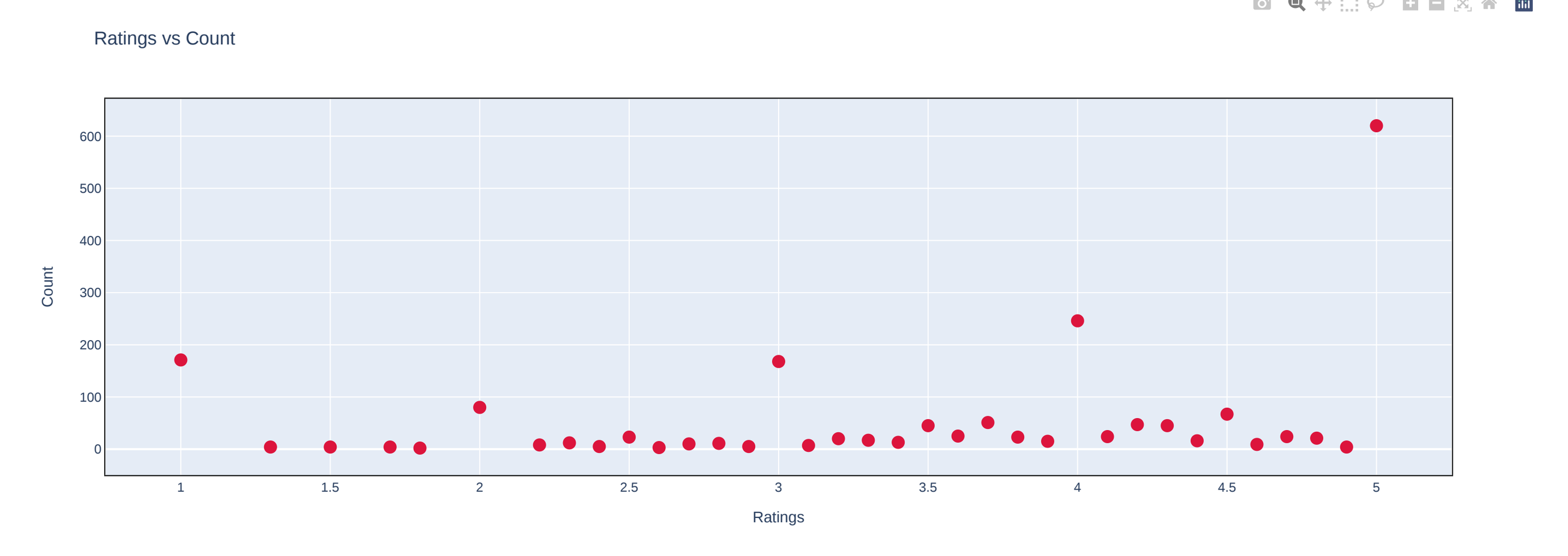
```python
# Filter for 5-star products
rating_5 = df[df['product_rating'] == '5']

# Count of main categories with 5-star ratings
top_product_type = rating_5['main_category'].value_counts()

# Count of brands with 5-star ratings
top_brand_type = rating_5['brand'].value_counts()

# Top 5 product categories
df_top_product = pd.DataFrame(top_product_type.head(5).reset_index())
df_top_product.columns = ['top_prod', 'count']

# Top 5 brands
df_top_brand = pd.DataFrame(top_brand_type.head(5).reset_index())
df_top_brand.columns = ['top_brands', 'count']

# Display the DataFrames
print(df_top_product)
print("-"*50)
print(df_top_brand)
```

```
           top_prod  count
0          Clothing    232
1         Jewellery     70
2          Footwear     47
3           Watches     47
4  Kitchen & Dining     37
--------------------------------------------------
  top_brands  count
0    Regular     21
1       Slim     13
2      Black      9
3      Bosch      9
4        JDX      8
```

Top-rated products and brands

In [22]:

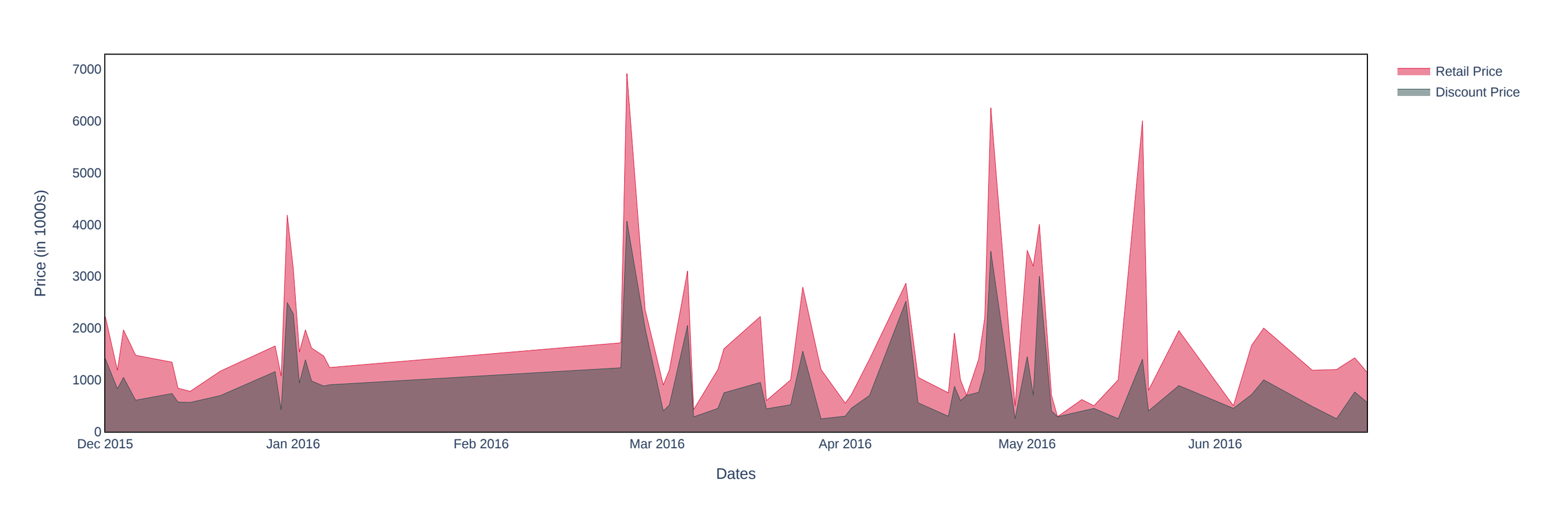
```python
# Concatenate the DataFrames horizontally
df_product_brand_rate5 = pd.concat([df_top_product, df_top_brand], axis=1)

# Remove rows with 'No rating available'
df.drop(df.index[df['product_rating'] == 'No rating available'], inplace=True)

# Count of each rating
ratings = pd.DataFrame(df['product_rating'].value_counts().reset_index())
ratings.columns = ['Ratings', 'Counts']

# Convert 'Ratings' to float for numerical sorting
ratings['Ratings'] = ratings['Ratings'].astype(float)

# Sort by 'Ratings' in descending order
ratings = ratings.sort_values(by=['Ratings'], ascending=[False])

# Plot the result
x = ratings['Ratings']
y = ratings['Counts']

figdot2 = go.Figure()
figdot2.add_trace(go.Scatter(
    x=x,
    y=y,
    marker=dict(color="crimson", size=12),
    mode="markers",
    name="ratings",
))

figdot2.update_layout(
    title="Ratings vs Count",
    xaxis_title="Ratings",
    yaxis_title="Count",
)

figdot2.update_xaxes(showline=True, linewidth=1, linecolor='black', mirror=True)
figdot2.update_yaxes(showline=True, linewidth=1, linecolor='black', mirror=True)

figdot2.show()
```



Visualize the trend of average retail and discounted prices over time using an area plot.

In [23]:

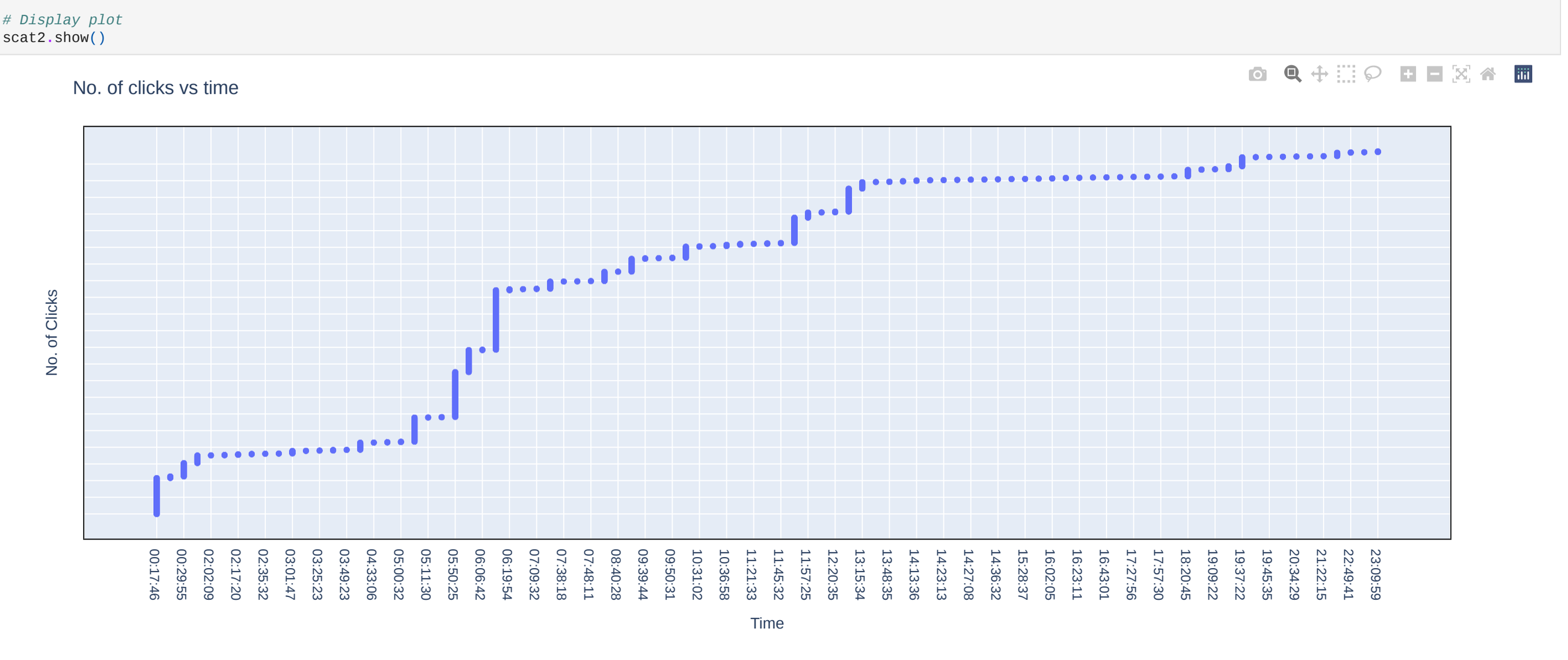
```python
# Group and aggregate data
df_date_retail = pd.DataFrame(df.groupby("date")[["retail_price"]].mean().reset_index())
df_date_discount = pd.DataFrame(df.groupby("date")[["discounted_price"]].mean().reset_index())

# Concatenate DataFrames
df_date_price = pd.concat([df_date_retail, df_date_discount], axis=1)

# Remove duplicate columns
df_date_price = df_date_price.loc[:, ~df_date_price.columns.duplicated()]

# Prepare data for plotting
x = df_date_price['date']
y1 = df_date_price['retail_price']
y2 = df_date_price['discounted_price']

# Create and customize plot
fig_area2 = go.Figure()

fig_area2.add_trace(go.Scatter(
    x=x,
    y=y1,
    fill='tozeroy',
    name='Retail Price',
    line=dict(width=0.5, color='crimson')
))

fig_area2.add_trace(go.Scatter(
    x=x,
    y=y2,
    fill='tozeroy',
    name='Discount Price',
    line=dict(width=0.5, color='darkslategray')
))

fig_area2.update_layout(
    xaxis_title="Dates",
    yaxis_title="Price (in 1000s)",
    plot_bgcolor='white'
)

fig_area2.update_xaxes(showline=True, linewidth=1, linecolor='black', mirror=True)
fig_area2.update_yaxes(showline=True, linewidth=1, linecolor='black', mirror=True)

fig_area2.show()
```



Visualize how the number of clicks on a product varies over time

In [24]:

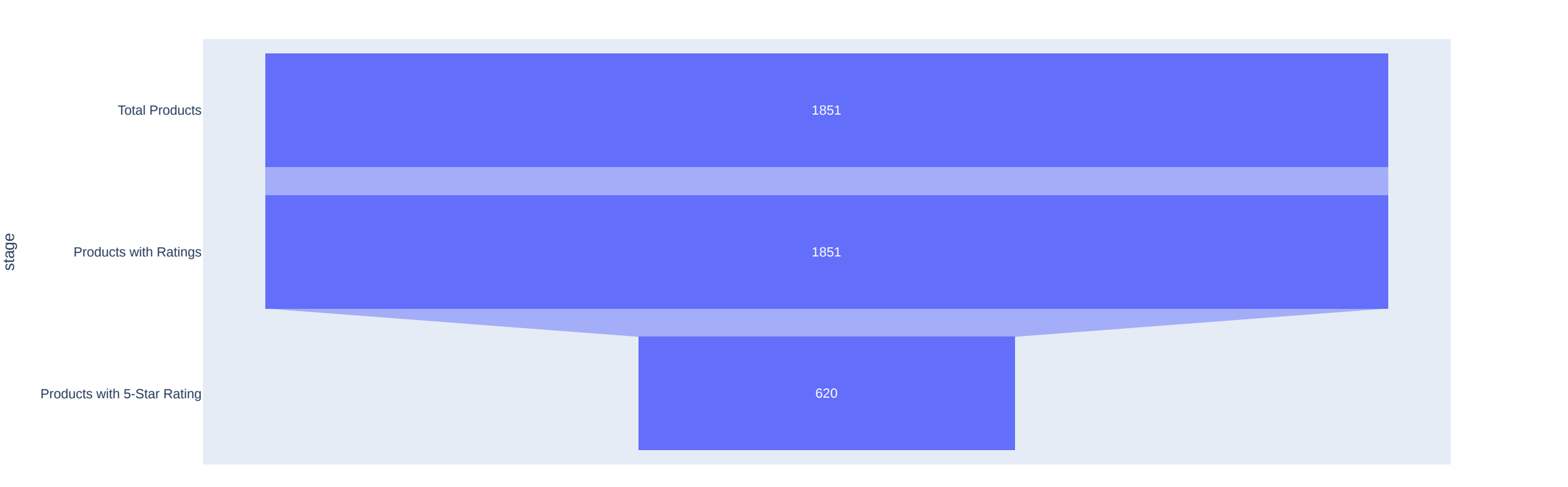
```python
# Create scatter plot
scat2 = px.scatter(x=df['Time'].sort_values(ascending=True), y=df['product_url'])

# Update layout
scat2.update_layout(
    title='No. of clicks vs time',  # Title of the plot
    xaxis_title='Time',  # X-axis label
    yaxis_title='No. of Clicks'  # Y-axis label
)

# Update axes
scat2.update_xaxes(showline=True, linewidth=1, linecolor='black', mirror=True)
scat2.update_yaxes(showline=True, linewidth=1, linecolor='black', mirror=True)

# Hide y-axis tick labels
scat2.update_yaxes(showticklabels=False)

# Display plot
scat2.show()
```



Total products to those receiving 5-star ratings at the end!

In [25]:

```python
# Calculate total products
total_prod = len(df['pid'])

# Calculate total rated products
total_ratings = len(df[df['product_rating'] != 'No rating available'])

# Calculate 5-star rated products
top_ratings = len(df[df['product_rating'] == '5'])

# Prepare data for funnel plot
df_funnel_1 = pd.DataFrame({
    'number': [total_prod, total_ratings, top_ratings],
    'stage': ["Total Products", "Products with Ratings", "Products with 5-Star Rating"]
})

# Create and display funnel plot
funnel_1_fig = px.funnel(df_funnel_1, x='number', y='stage')
funnel_1_fig.show()
```



In [ ]: