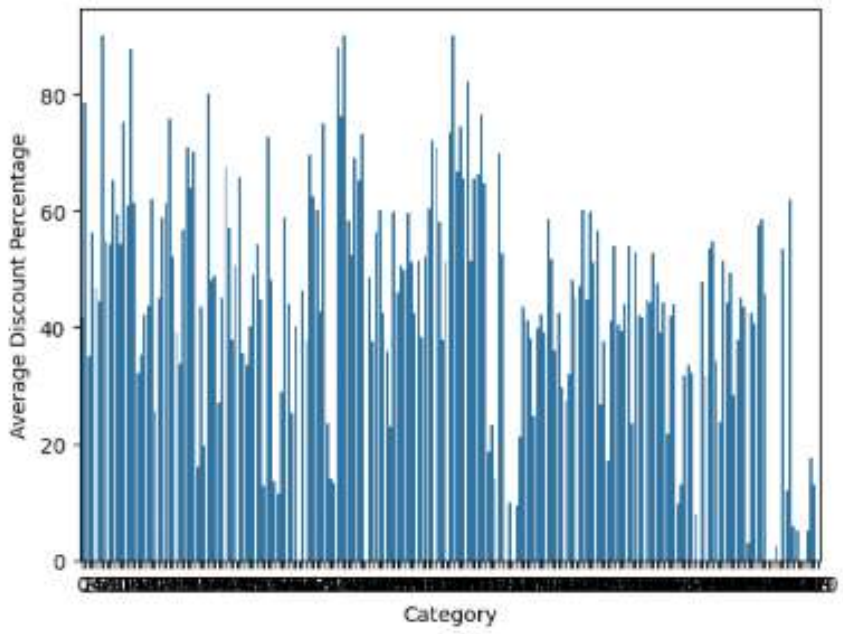
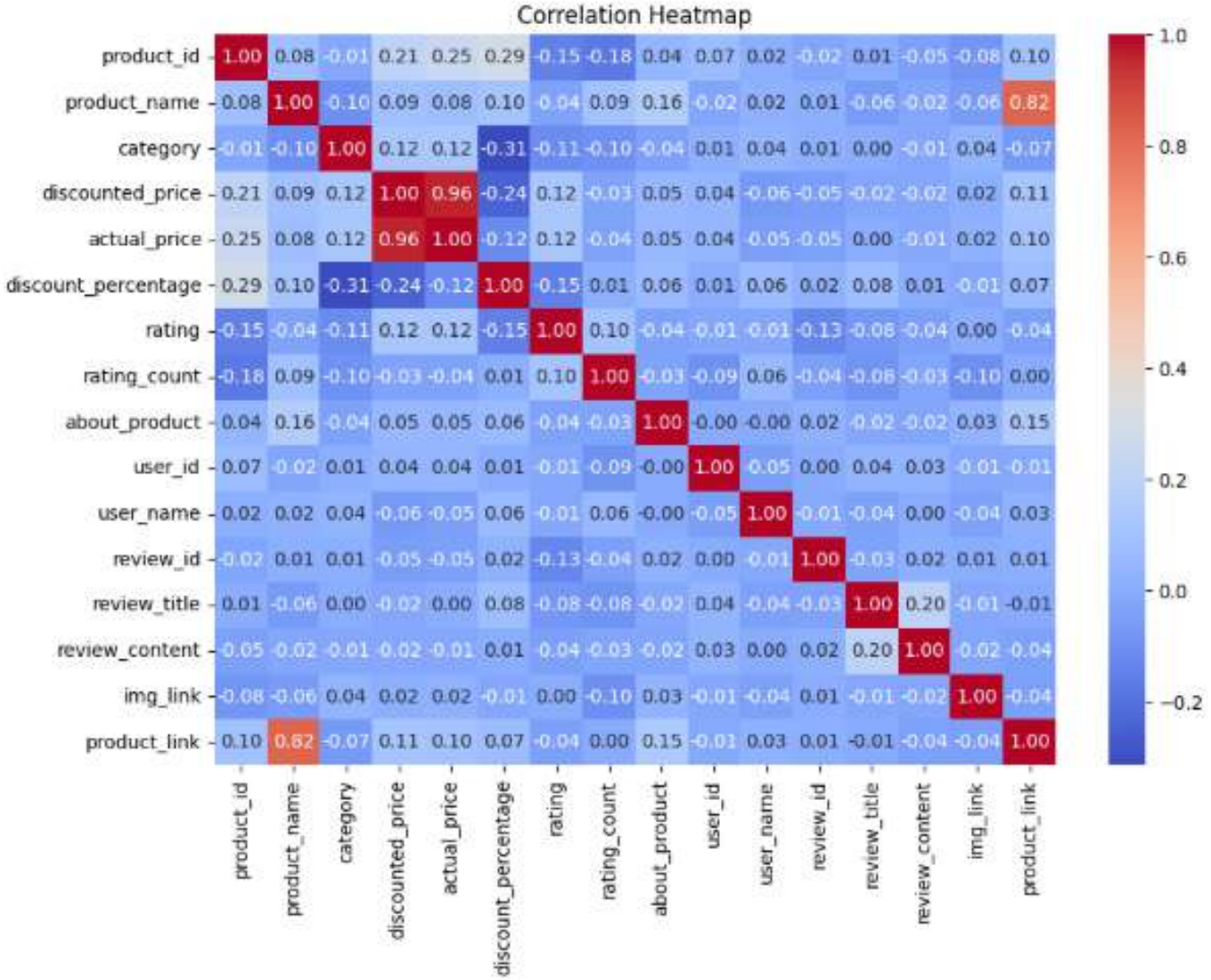
```python
# Calculate average discount percentage per category
avg_discount_per_category = df1.groupby('category')['discount_percentage'].mean()

# Display results
#print(avg_discount_per_category)

# Optional: Visualization
sns.barplot(x=avg_discount_per_category.index, y=avg_discount_per_category.values)
plt.xlabel("Category")
plt.ylabel("Average Discount Percentage")
plt.show()
```



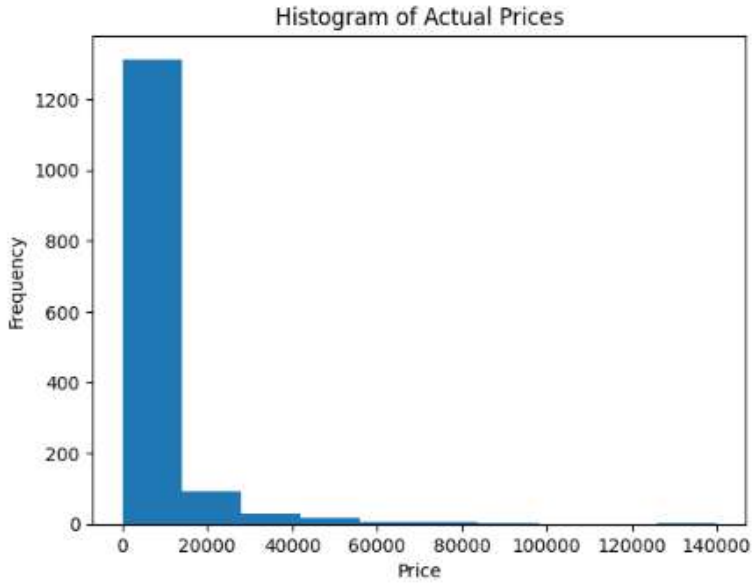
Average discount percentages vary widely across categories, ranging from 0% to 78.39%. Categories 1 and 3 stand out with notably higher average discounts (78.39% and 56.34%), suggesting potential factors like clearance efforts, high competition, or lower-profit margins. Categories 0, 206, 207, 210 have average discounts of 0%, indicating consistent pricing or strong demand for products within those categories. Other categories exhibit varying discount percentages, likely reflecting diverse pricing strategies and market dynamics.


Correlation Heatmap
product_id 1.00 0.08 -0.01 0.21 0.25 0.29 -0.15 -0.18 0.04 0.07 0.02 -0.02 0.01 -0.05 -0.08 0.10
product_name 0.08 1.00 -0.10 0.09 0.08 0.10 0.04 0.09 0.16 -0.02 0.02 0.01 -0.06 -0.02 -0.06 0.82
category -0.01 -0.10 1.00 0.12 0.12 -0.31 -0.11 -0.10 -0.04 0.01 0.04 0.01 0.00 -0.01 0.04 -0.07
discounted_price 0.21 0.09 0.12 1.00 0.96 -0.24 0.12 -0.03 0.05 0.04 -0.06 -0.05 -0.02 -0.02 0.02 0.11
actual_price 0.25 0.08 0.12 0.96 1.00 -0.12 0.12 -0.04 0.05 0.04 -0.05 -0.05 0.00 -0.01 0.02 0.10
discount_percentage 0.29 0.10 -0.31 -0.24 -0.12 1.00 -0.15 0.01 0.06 0.01 0.06 0.02 0.08 0.01 -0.01 0.07
rating -0.15 -0.04 -0.11 0.12 0.12 -0.15 1.00 0.10 -0.04 -0.01 -0.01 -0.13 -0.08 -0.04 0.00 -0.04
rating_count -0.18 0.09 -0.10 -0.03 -0.04 0.01 0.10 1.00 -0.03 -0.09 0.06 -0.04 -0.06 -0.03 -0.10 0.00
about_product 0.04 0.16 -0.04 0.05 0.05 0.06 -0.04 -0.03 1.00 -0.00 -0.00 0.02 -0.02 -0.02 0.03 0.15
user_id 0.07 -0.02 0.01 0.04 0.04 0.01 -0.01 -0.09 -0.00 1.00 -0.05 0.00 0.04 0.03 -0.01 -0.01
user_name 0.02 0.02 0.04 -0.06 -0.05 0.06 -0.01 0.06 -0.00 -0.05 1.00 -0.01 -0.04 0.00 -0.04 0.03
review_id -0.02 0.01 0.01 -0.05 -0.05 0.02 -0.13 -0.04 0.02 0.00 -0.01 1.00 -0.03 0.02 0.01 0.01
review_title 0.01 -0.06 0.00 -0.02 0.00 0.08 -0.08 -0.08 -0.02 0.04 -0.04 -0.03 1.00 0.20 -0.01 -0.01
review_content -0.05 -0.02 -0.01 -0.02 -0.01 0.01 0.04 -0.03 -0.02 0.03 0.00 0.02 0.20 1.00 -0.02 -0.04
img_link -0.08 -0.06 0.04 0.02 0.02 -0.01 0.00 -0.10 0.03 -0.01 -0.04 0.01 -0.01 -0.02 1.00 -0.04
product_link 0.10 0.82 -0.07 0.11 0.10 0.07 -0.04 0.00 0.15 -0.01 0.03 0.01 -0.01 -0.04 -0.04 1.00
product_id
product_name
category
discounted_price
actual_price
discount_percentage
rating
rating_count
about_product
user_id
user_name
review_id
review_title
review_content
img_link
product_link
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

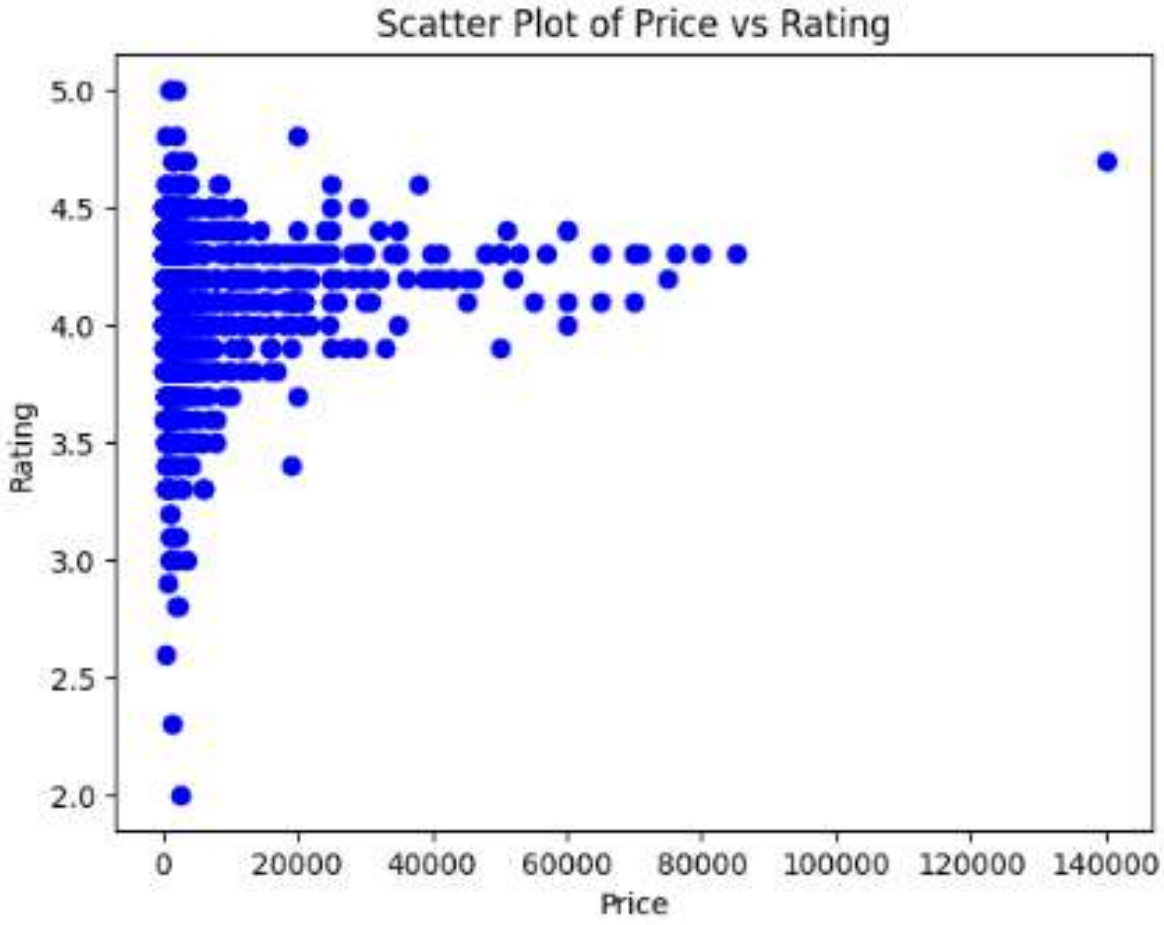
```
df1.corr()
```

	product_id	product_name	category	discounted_price	actual_price	discount_percentage	rating	rating_count	about_product	user_id	user_name	review_id	review_title	review_content	img_link	product_link
product_id	1.000000	0.084089	-0.012565	0.206448	0.246733	0.289514	-0.149105	-0.175530	0.041404	0.065688	0.016145	-0.024282	0.007650	-0.046273	-0.078803	0.096205
product_name	0.084089	1.000000	-0.103778	0.090665	0.078567	0.101913	-0.035592	0.092450	0.158263	-0.024093	0.024598	0.013492	-0.060594	-0.018505	-0.060858	0.823725
category	-0.012565	-0.103778	1.000000	0.119365	0.122451	-0.314465	-0.109424	-0.098421	-0.038753	0.012707	0.037822	0.014015	0.004712	-0.012107	0.038850	-0.067710
discounted_price	0.206448	0.090665	0.119365	1.000000	0.961915	-0.242412	0.120386	-0.027081	0.052618	0.041731	-0.063069	-0.049757	-0.020981	-0.015904	0.018917	0.110186
actual_price	0.246733	0.078567	0.122451	0.961915	1.000000	-0.118098	0.121744	-0.035959	0.048529	0.041501	-0.049567	-0.045640	0.004521	-0.013948	0.020080	0.099286
discount_percentage	0.289514	0.101913	-0.314465	-0.242412	-0.118098	1.000000	-0.154563	0.011097	0.060846	0.008288	0.064618	0.022442	0.076043	0.005993	-0.011701	0.074548
rating	-0.149105	-0.035592	-0.109424	0.120386	0.121744	-0.154563	1.000000	0.101700	-0.036056	-0.014528	-0.012704	-0.134071	-0.075715	-0.044567	0.003993	-0.044125
rating_count	-0.175530	0.092450	-0.098421	-0.027081	-0.035959	0.011097	0.101700	1.000000	-0.030821	-0.094512	0.059160	-0.043089	-0.084239	-0.025268	-0.100781	0.003665
about_product	0.041404	0.158263	-0.038753	0.052618	0.048529	0.060846	-0.036056	-0.030821	1.000000	-0.001946	-0.004547	0.019690	-0.015824	-0.024567	0.030874	0.147208
user_id	0.065688	-0.024093	0.012707	0.041731	0.041501	0.008288	-0.014528	-0.094512	-0.001946	1.000000	-0.051368	0.001685	0.039035	0.031295	-0.014114	-0.008662
user_name	0.016145	0.024598	0.037822	-0.063069	-0.049567	0.064618	-0.012704	0.059160	-0.004547	-0.051368	1.000000	-0.009057	-0.038880	0.002433	-0.038779	0.026996
review_id	-0.024282	0.013492	0.014015	-0.049757	-0.045640	0.022442	-0.134071	-0.043089	0.019690	0.001685	-0.009057	1.000000	-0.034569	0.016665	0.005422	0.014423
review_title	0.007650	-0.060594	0.004712	-0.020981	0.004521	0.076043	-0.075715	-0.084239	-0.015824	0.039035	-0.038880	-0.034569	1.000000	0.201131	-0.012056	-0.006080
review_content	-0.046273	-0.018505	-0.012107	-0.015904	-0.013948	0.005993	-0.044567	-0.025268	-0.024567	0.031295	0.002433	0.016665	0.201131	1.000000	-0.020283	-0.038574
img_link	-0.078803	-0.060858	0.038850	0.018917	0.020080	-0.011701	0.003993	-0.100781	0.030874	-0.014114	-0.038779	0.005422	-0.012056	-0.020283	1.000000	-0.043935
product_link	0.096205	0.823725	-0.067710	0.110186	0.099286	0.074548	-0.044125	0.003665	0.147208	-0.008662	0.026996	0.014423	-0.006080	-0.038574	-0.043935	1.000000

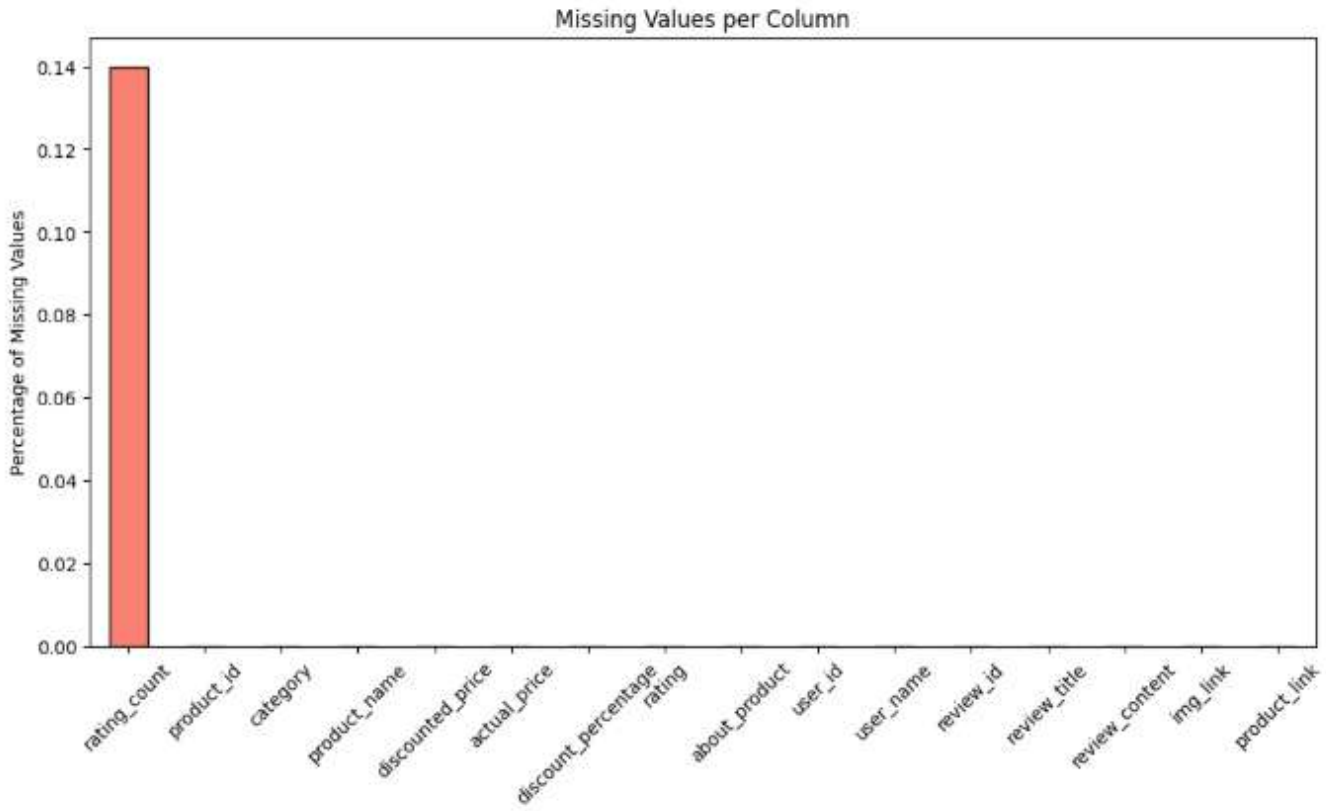
```
plt.hist(df1['actual_price'])
plt.title('Histogram of Actual Prices')
plt.xlabel('Price')
plt.ylabel('Frequency')
plt.show()
```



```
plt.scatter( df1['actual_price'],df1['rating'], color='blue')
plt.title('Scatter Plot of Price vs Rating')
plt.ylabel('Rating')
plt.xlabel('Price')
plt.show()
```



```
plt.figure(figsize=(12,6))
per.plot(kind='bar', color='salmon', edgecolor='black')
plt.ylabel("Percentage of Missing Values")
plt.title("Missing Values per Column")
plt.xticks(rotation=45)
plt.show()
```



```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(22, 10))
plt.title("Null values in each column")
sns.heatmap(df.isnull(), yticklabels=False, cbar=False, cmap='viridis')
plt.show()
```