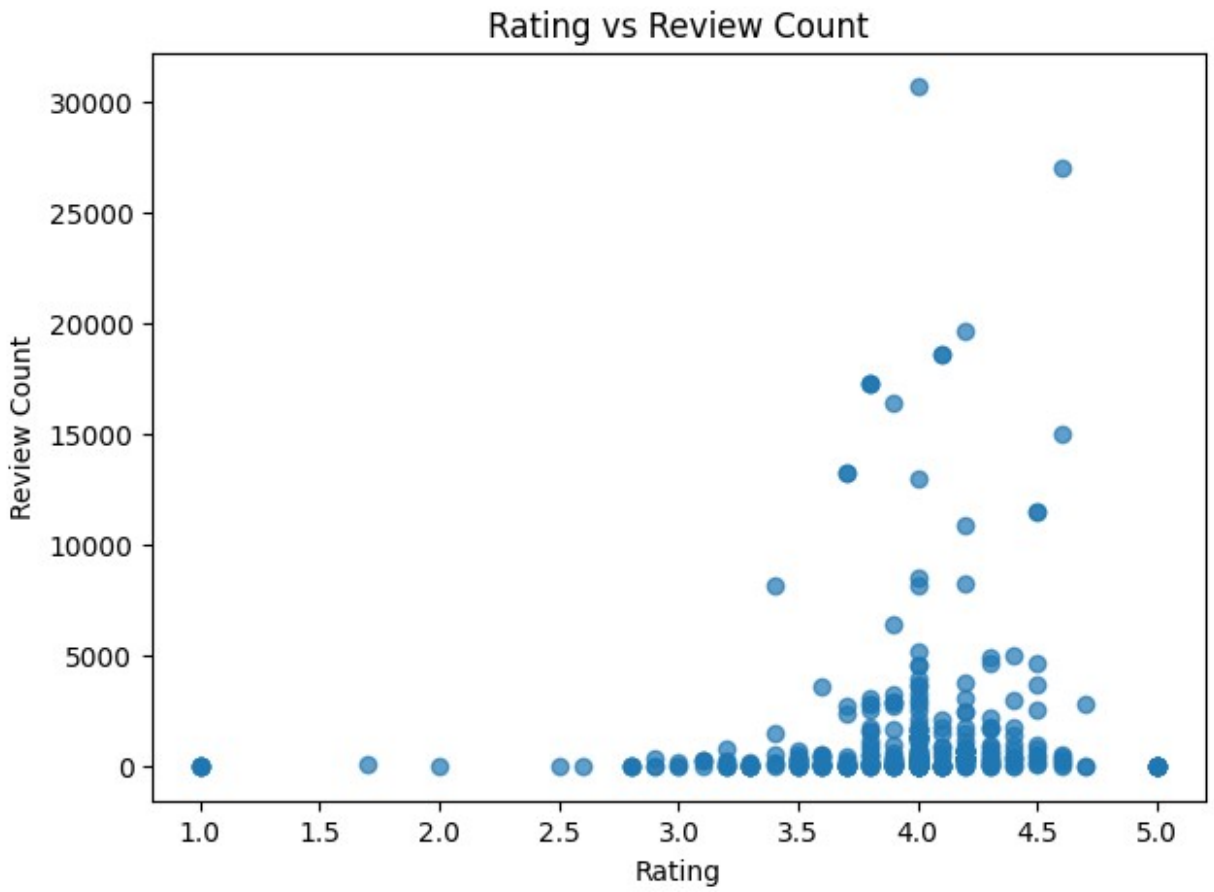
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('cleaned_tvsamazon.csv')  # replace with actual
filename
df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 403,\n  \"fields\": [\
n    {\n      \"column\": \"listed_price\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 10829.89595984074,\n
\"min\": 80.0,\n        \"max\": 66990.0,\n
\"num_unique_values\": 204,\n        \"samples\": [\n
21990.0,\n          38490.0,\n          3100.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"delivery_date\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 8,\n        \"samples\": [\n          \"30 Dec,
2025 - 1 Jan\",\n          \"Fri, 26 Dec\",\n          \"Sat, 20
Dec\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"review_count\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 3570.110299591765,\n        \"min\":
0.0,\n        \"max\": 30700.0,\n        \"num_unique_values\": 208,\n
\"samples\": [\n          14.0,\n          148.0,\n          171.0\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"product_name\",\n
\"properties\": {\n        \"dtype\": \"string\",\n
\"num_unique_values\": 330,\n        \"samples\": [\n          \"Sony
108 cm (43 inches) BRAVIA 2M2 Series 4K Ultra HD Smart LED Google TV
K-43S22BM2\",\n          \"Acko 1 Year Extended Warranty for TVs from
Rs 10,001 to Rs 15,000 (Email Delivery)\",\n          \"Xiaomi 80 cm
(32 inches) G QLED Series Smart TV L32MB-APIN\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"mrp\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 152367.2351533869,\n
\"min\": 8.45,\n        \"max\": 2999000.0,\n
\"num_unique_values\": 168,\n        \"samples\": [\n
1299.0,\n          41400.0,\n          1599.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"rating\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\":
0.5582488612122525,\n        \"min\": 1.0,\n        \"max\": 5.0,\n
\"num_unique_values\": 26,\n        \"samples\": [\n          4.6,\n
5.0,\n          4.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"discount_percentage\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 101.06874186644495,\n        \"min\": -
1900.0,\n        \"max\": 99.0,\n        \"num_unique_values\": 145,\n
\"samples\": [\n          68.0,\n          78.68,\n          62.03\n
```

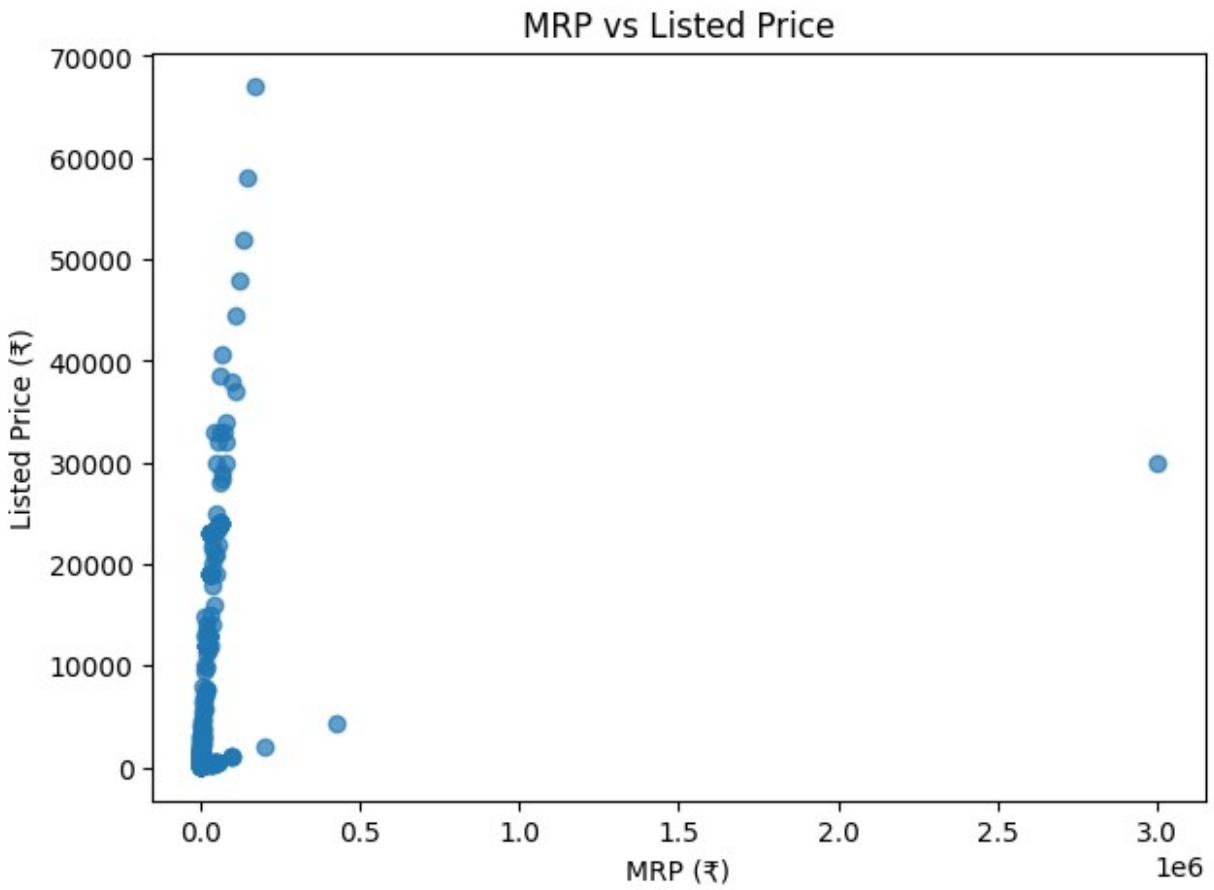
```
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    }\n  ]\n}","type":"dataframe","variable_name":"df"}
```

```
plt.figure(figsize=(7,5))
plt.scatter(df['listed_price'], df['discount_percentage'], alpha=0.7)
plt.xlabel('Listed Price (₹)')
plt.ylabel('Discount Percentage')
plt.title('Listed Price vs Discount Percentage')
plt.show()
```

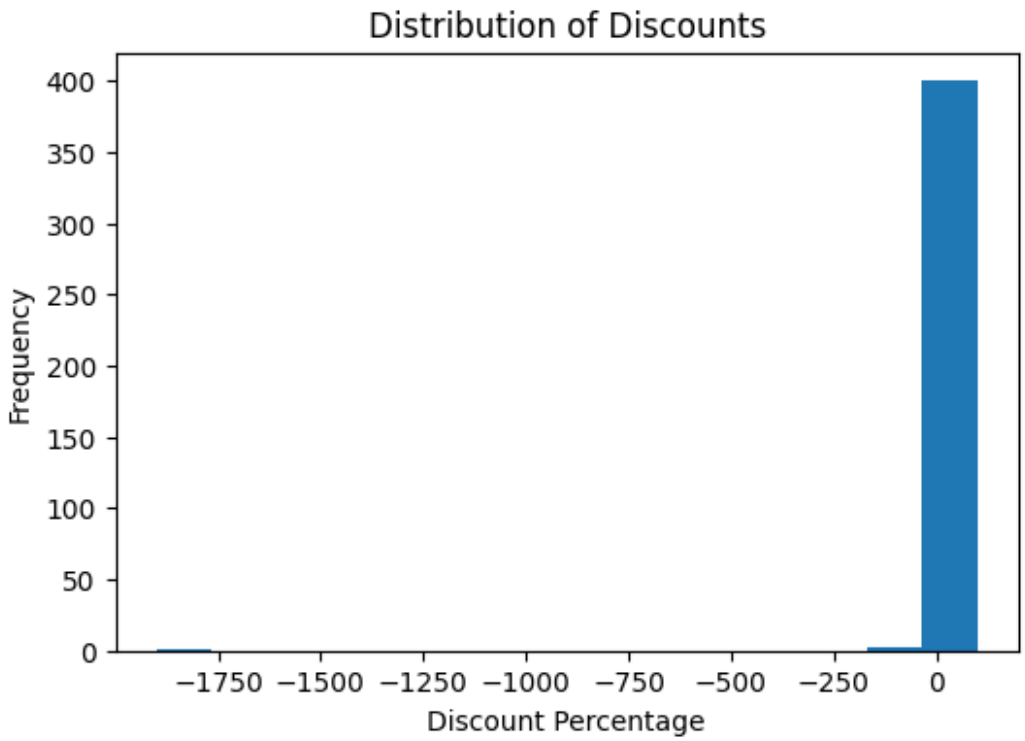

```
plt.figure(figsize=(7,5))
plt.scatter(df['rating'], df['review_count'], alpha=0.7)
plt.xlabel('Rating')
plt.ylabel('Review Count')
plt.title('Rating vs Review Count')
plt.show()
```



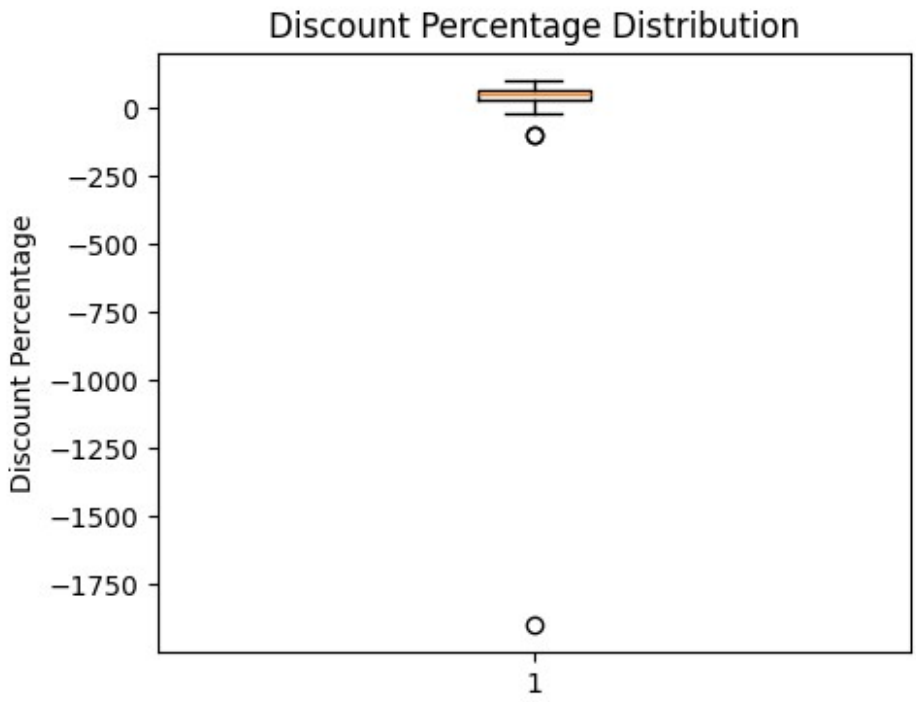

```
plt.figure(figsize=(7,5))
plt.scatter(df['mrp'], df['listed_price'], alpha=0.7)
plt.xlabel('MRP (₹)')
plt.ylabel('Listed Price (₹)')
plt.title('MRP vs Listed Price')
plt.show()
```



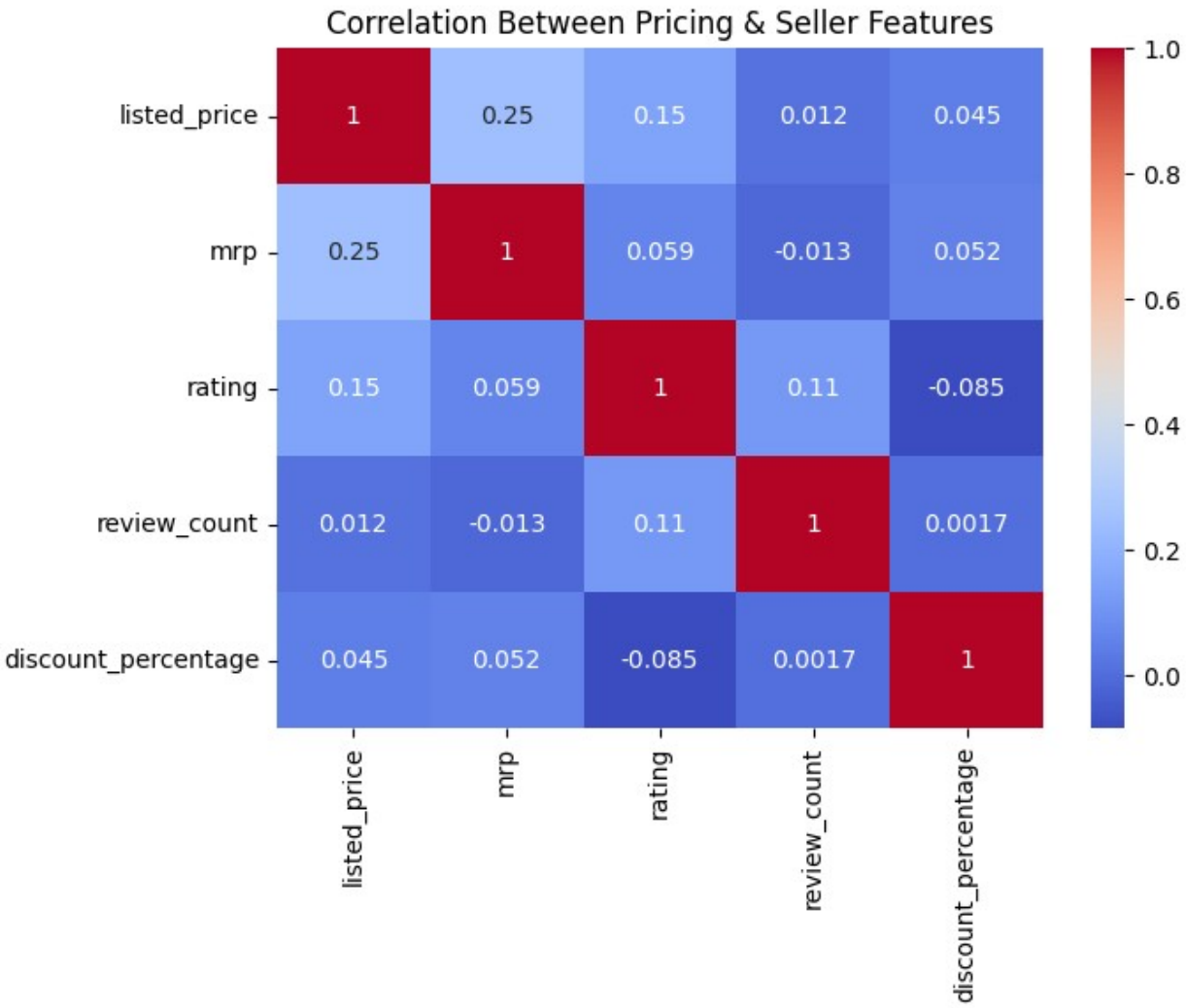

```
plt.figure(figsize=(6,4))
plt.hist(df['discount_percentage'], bins=15)
plt.xlabel('Discount Percentage')
plt.ylabel('Frequency')
plt.title('Distribution of Discounts')
plt.show()
```




```
plt.figure(figsize=(5,4))
plt.boxplot(df['discount_percentage'])
plt.ylabel('Discount Percentage')
plt.title('Discount Percentage Distribution')
plt.show()
```




```
plt.figure(figsize=(7,5))
sns.heatmap(

df[['listed_price','mrp','rating','review_count','discount_percentage'
]].corr(),
    annot=True,
    cmap='coolwarm'
)
plt.title('Correlation Between Pricing & Seller Features')
plt.show()
```


Correlation Between Pricing & Seller Features
listed_price
mrp
rating
review_count
discount_percentage
1
0.25
0.15
0.012
0.045
0.25
1
0.059
-0.013
0.052
0.15
0.059
1
0.11
-0.085
0.012
-0.013
0.11
1
0.0017
0.045
0.052
-0.085
0.0017
1
1.0
0.8
0.6
0.4
0.2
0.0