In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv('googleplaystore.csv')
data.head()
```

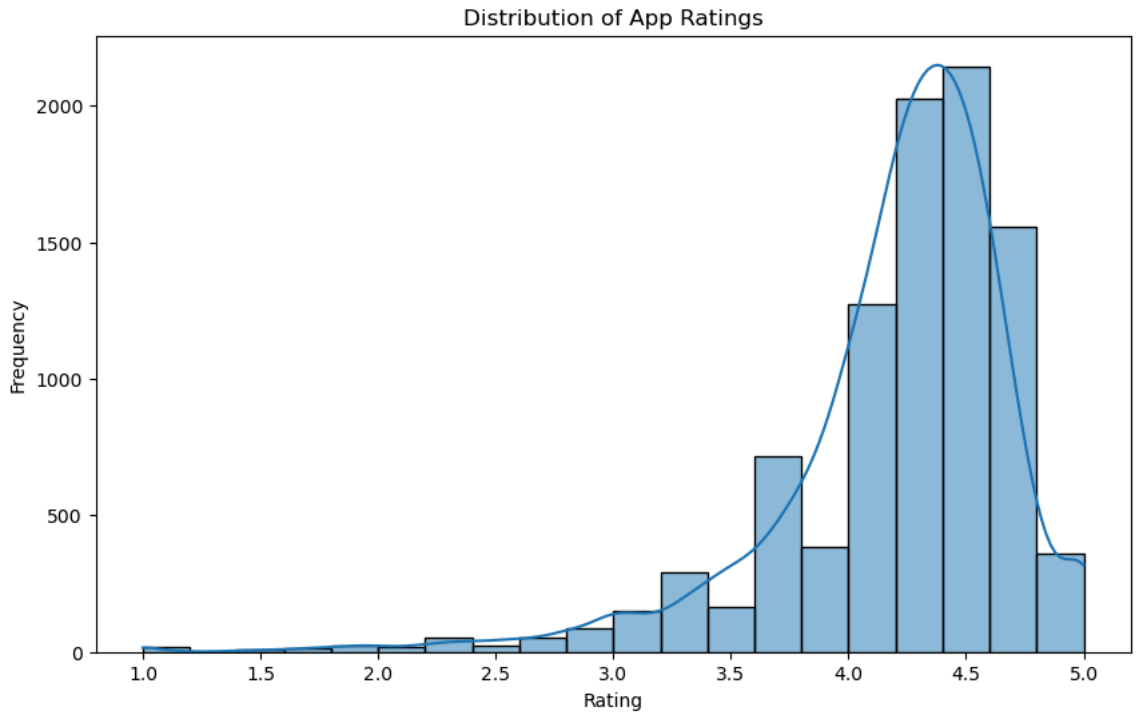
Out[1]:

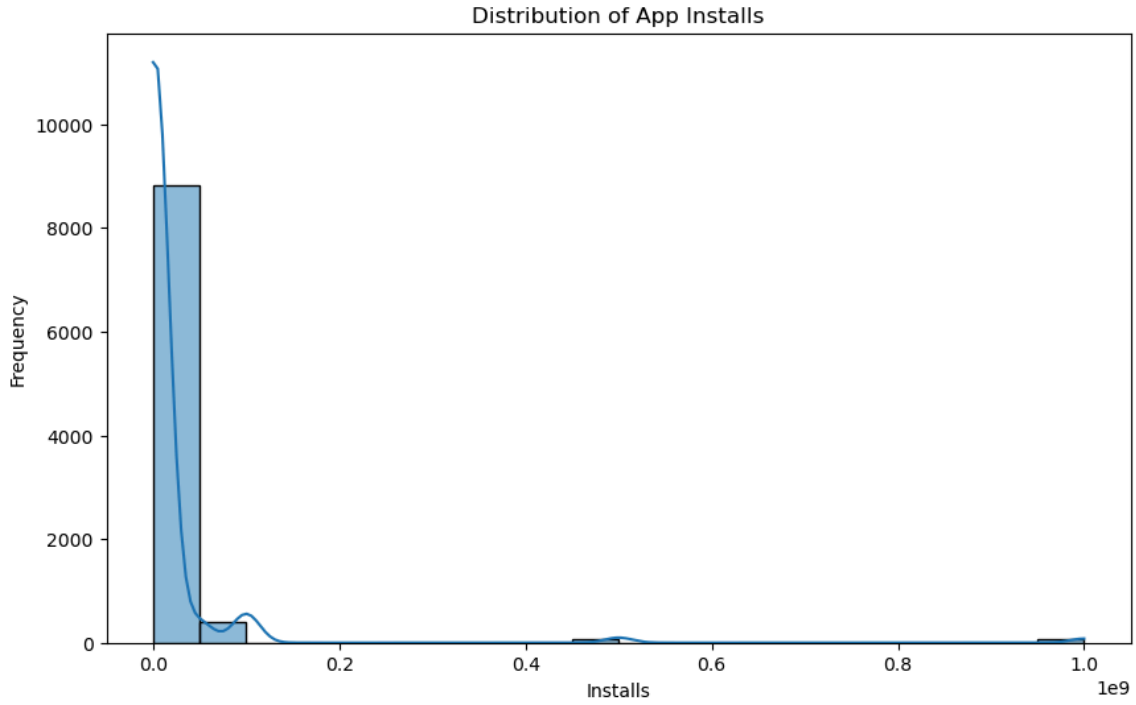
	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone

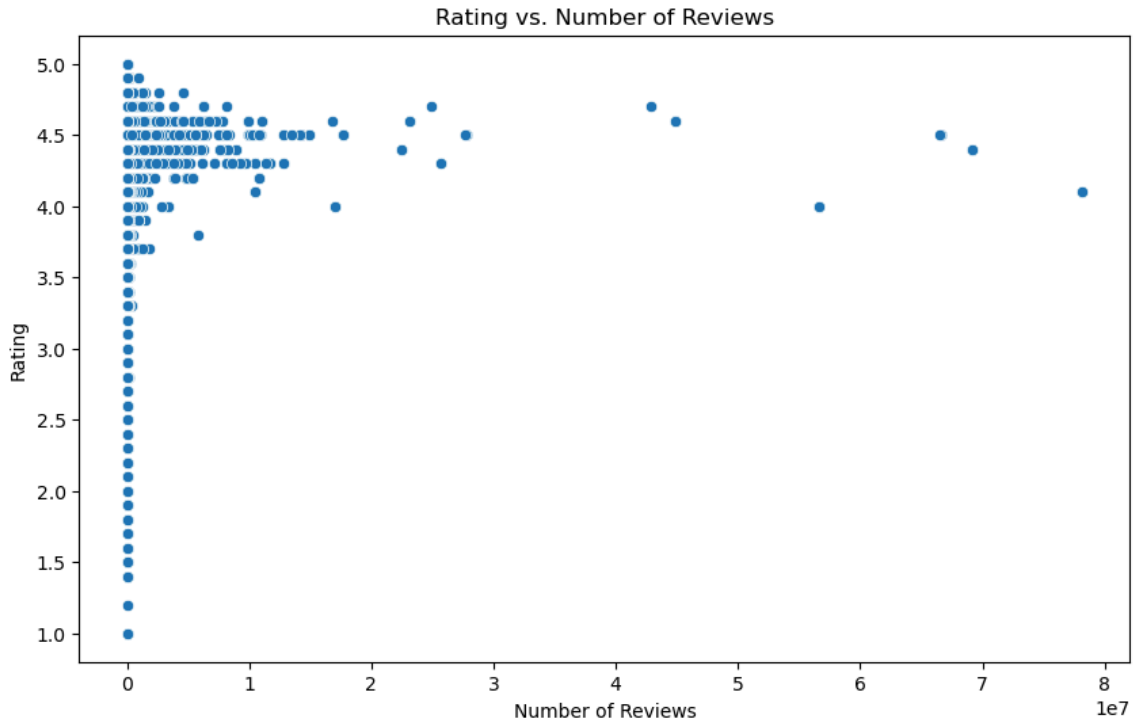
In [2]:
```
data.isnull().sum()
data = data.dropna()
data['Reviews'] = data['Reviews'].astype(int)
data['Installs'] = data['Installs'].str.replace('+', '').str.replace(',', '').astype
data['Price'] = data['Price'].str.replace('$', '').astype(float)

def convert_size(size):
    if 'M' in size:
        return float(size.replace('M', ''))
    elif 'k' in size:
        return float(size.replace('k', '')) / 1000
    else:
        return np.nan

data['Size'] = data['Size'].apply(convert_size)
data['Category'] = data['Category'].astype('category')
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9360 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   App             9360 non-null   object
 1   Category        9360 non-null   category
 2   Rating          9360 non-null   float64
 3   Reviews         9360 non-null   int32
 4   Size            7723 non-null   float64
 5   Installs        9360 non-null   int32
 6   Type            9360 non-null   object
 7   Price           9360 non-null   float64
 8   Content Rating  9360 non-null   object
 9   Genres          9360 non-null   object
 10  Last Updated    9360 non-null   object
 11  Current Ver     9360 non-null   object
 12  Android Ver     9360 non-null   object
dtypes: category(1), float64(3), int32(2), object(7)
memory usage: 887.9+ KB

C:\Users\ethan\AppData\Local\Temp\ipykernel_11412\2598140529.py:10: FutureWarnin
g: The default value of regex will change from True to False in a future version.
In addition, single character regular expressions will *not* be treated as litera
l strings when regex=True.
  data['Installs'] = data['Installs'].str.replace('+', '').str.replace(',', '').a
stype(int)
C:\Users\ethan\AppData\Local\Temp\ipykernel_11412\2598140529.py:11: FutureWarnin
g: The default value of regex will change from True to False in a future version.
In addition, single character regular expressions will *not* be treated as litera
l strings when regex=True.
  data['Price'] = data['Price'].str.replace('$', '').astype(float)
```

In [3]:
```
data.describe()

plt.figure(figsize=(10, 6))
sns.histplot(data['Rating'].dropna(), bins=20, kde=True)
plt.title('Distribution of App Ratings')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.show()

plt.figure(figsize=(10, 6))
sns.histplot(data['Installs'], bins=20, kde=True)
plt.title('Distribution of App Installs')
plt.xlabel('Installs')
plt.ylabel('Frequency')
plt.show()

plt.figure(figsize=(10, 6))
sns.scatterplot(x='Reviews', y='Rating', data=data)
plt.title('Rating vs. Number of Reviews')
plt.xlabel('Number of Reviews')
plt.ylabel('Rating')
plt.show()
```

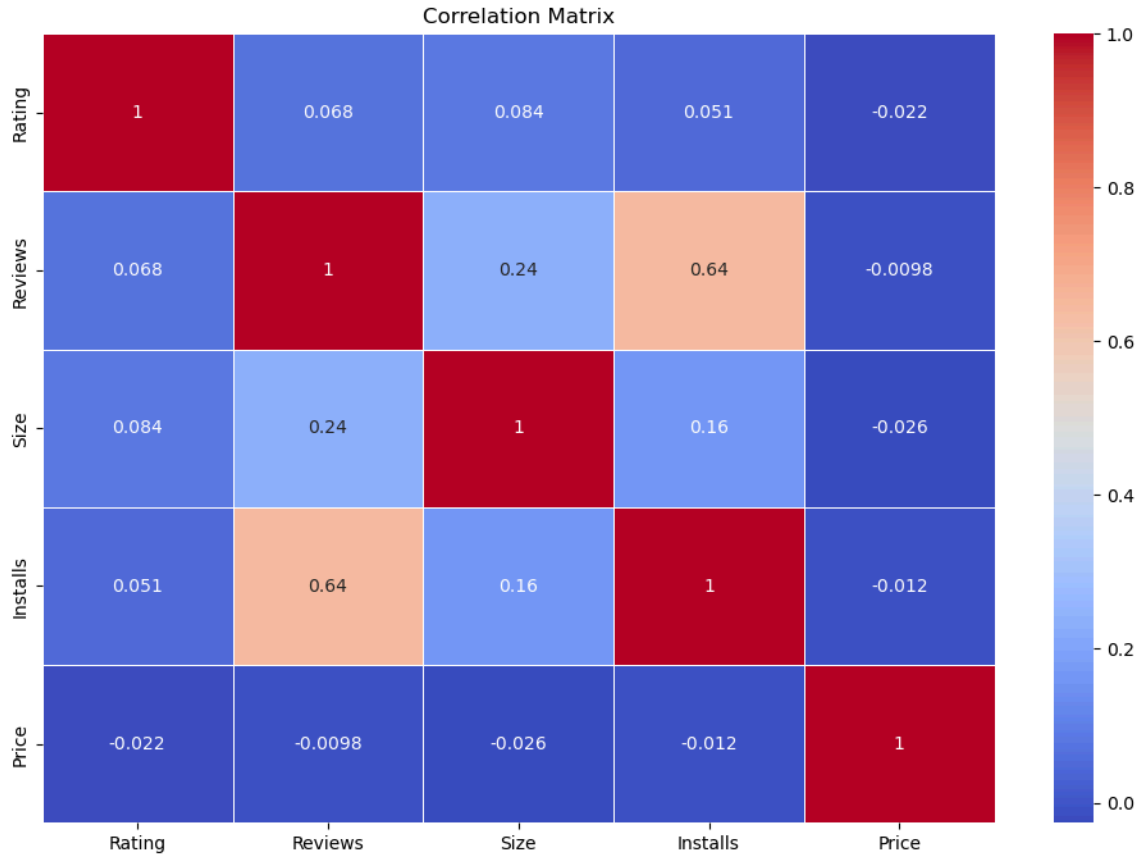



Distribution of App Installs
Frequency
10000
8000
6000
4000
2000
0
0.0
0.2
0.4
0.6
0.8
1.0
Installs
1e9


Rating vs. Number of Reviews
Rating
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0
1
2
3
4
5
6
7
8
Number of Reviews
1e7

In [4]:
```
correlation_matrix = data.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Matrix')
plt.show()
```

```
C:\Users\ethan\AppData\Local\Temp\ipykernel_11412\1098027073.py:2: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future ve
rsion, it will default to False. Select only valid columns or specify the value o
f numeric_only to silence this warning.
  correlation_matrix = data.corr()
```



In [ ]:
```
```