Import Libraries

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.impute import SimpleImputer

```

Load and Inspect Dataset

```
df = pd.read_csv("/content/AB_NYC_2019.csv")
print("Columns available:", df.columns)
df.head()

```

```
Columns available: Index(['id', 'name', 'host_id', 'host_name', 'neighbourhood_group',
       'neighbourhood', 'latitude', 'longitude', 'room_type', 'price',
       'minimum_nights', 'number_of_reviews', 'last_review',
       'reviews_per_month', 'calculated_host_listings_count',
       'availability_365'],
      dtype='object')
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_n
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	

Next steps: Generate code with `df` View recommended plots New interactive sheet

Standardize Column Names

```
df.columns = [col.capitalize() for col in df.columns]

```

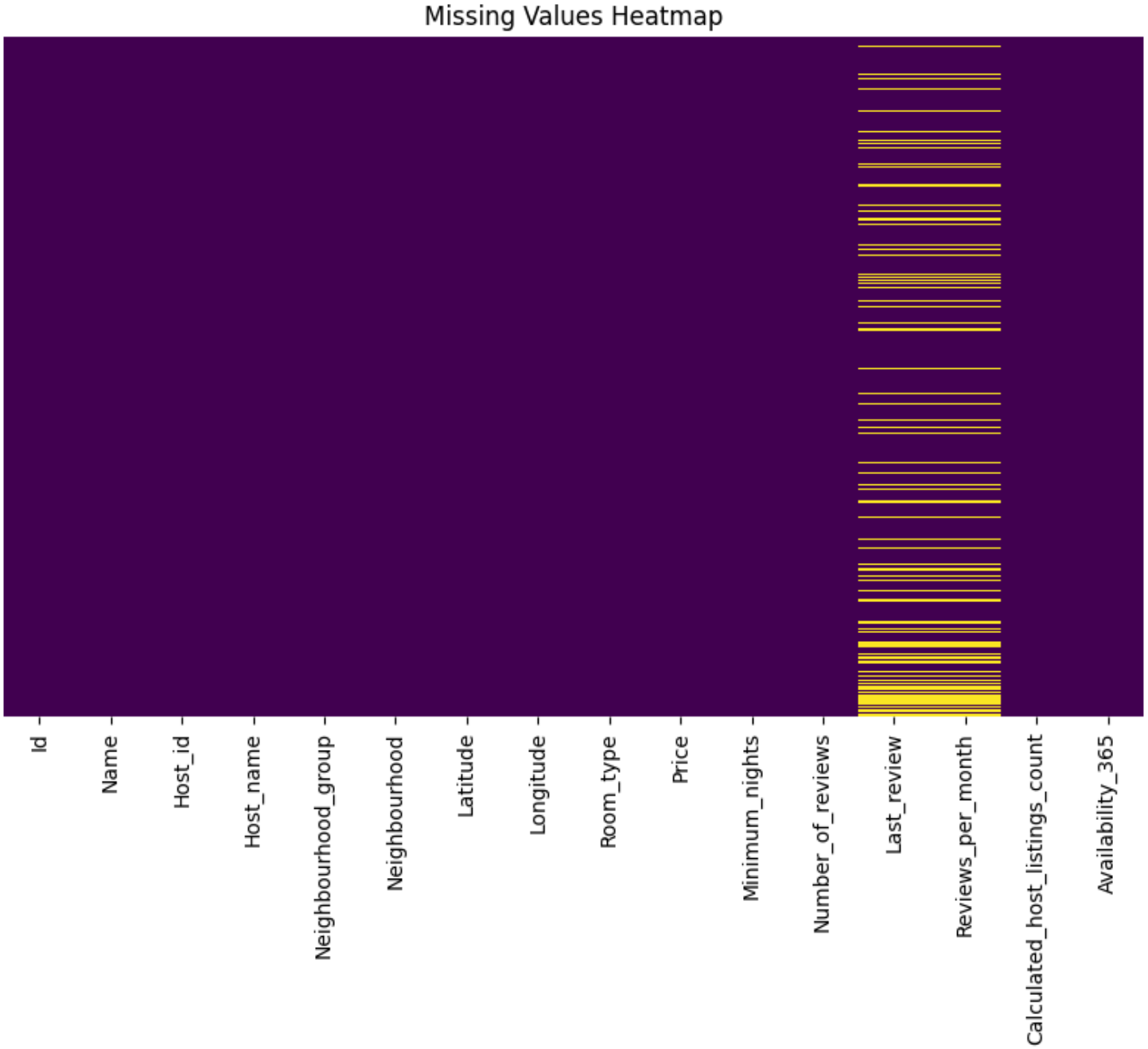
Visualize Missing Data

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

```
Id                                0
Name                             16
Host_id                           0
Host_name                        21
Neighbourhood_group               0
Neighbourhood                     0
Latitude                          0
Longitude                         0
Room_type                         0
Price                             0
Minimum_nights                    0
Number_of_reviews                 0
Last_review                   10052
Reviews_per_month             10052
Calculated_host_listings_count    0
Availability_365                  0
dtype: int64
```

```
plt.figure(figsize=(10,6))
sns.heatmap(df.isnull(), cbar=False, yticklabels=False, cmap="viridis")
plt.title("Missing Values Heatmap")
plt.show()

```



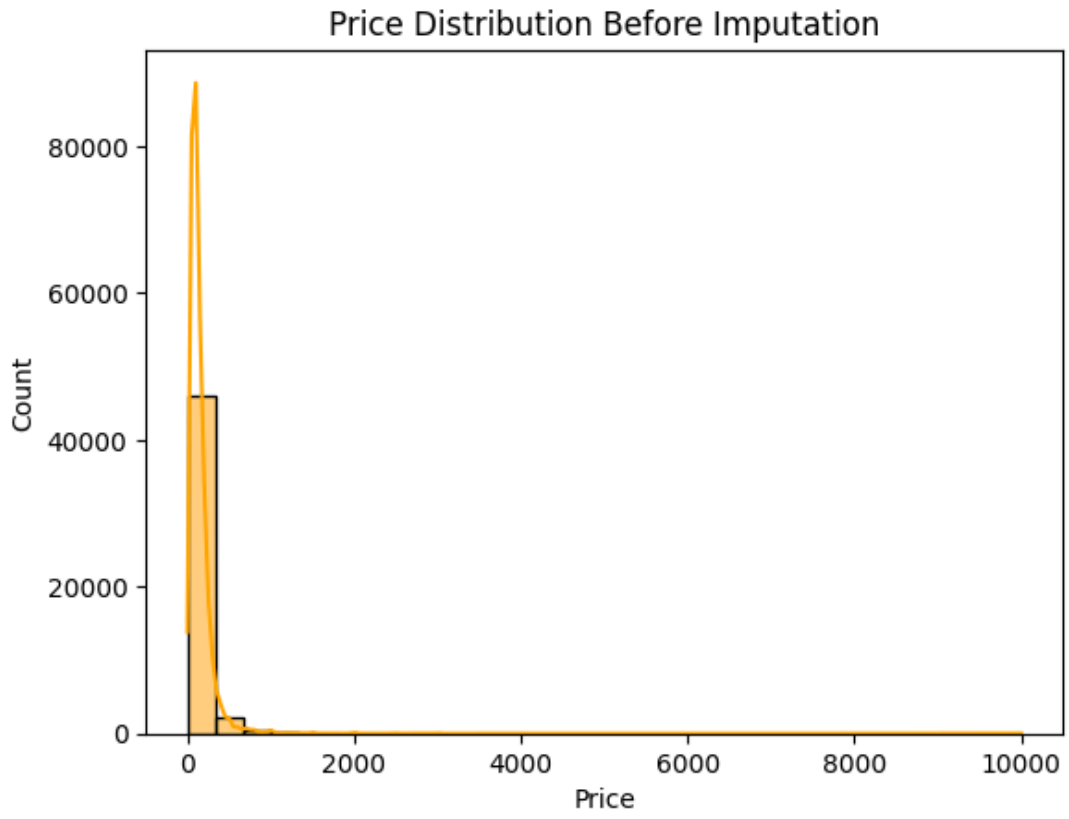
Drop the 'Cabin' column due to excessive missingness.

```
if "Cabin" in df.columns:
    df.drop(columns=["Cabin"], inplace=True)

```

Impute missing values in the 'Age' column with its median value.

```
#Handle Numerical Missing Values (Age)
sns.histplot(df['Price'], bins=30, kde=True, color="orange")
plt.title("Price Distribution Before Imputation")
plt.show()
```

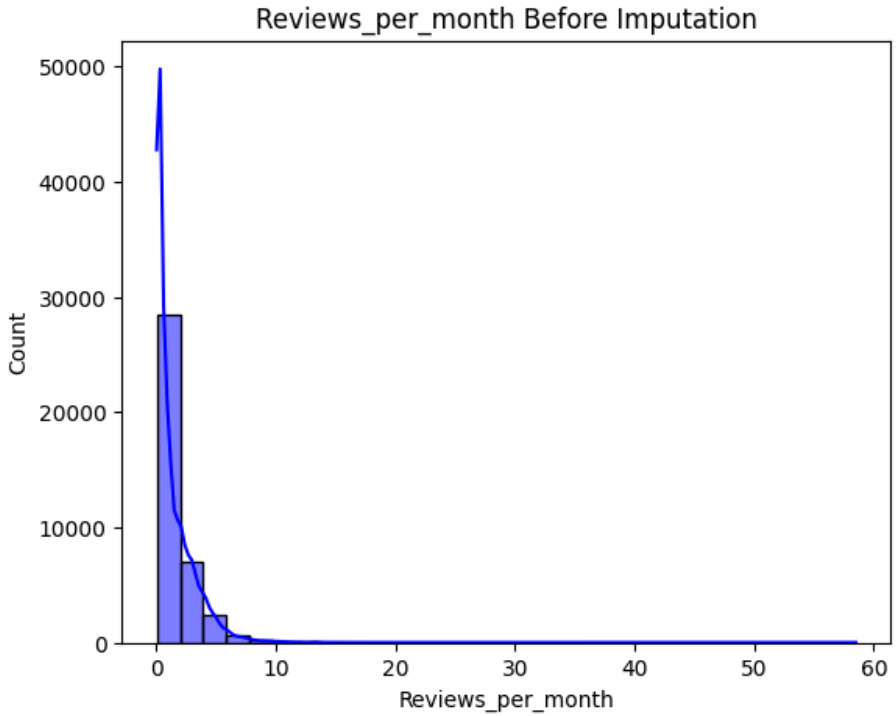


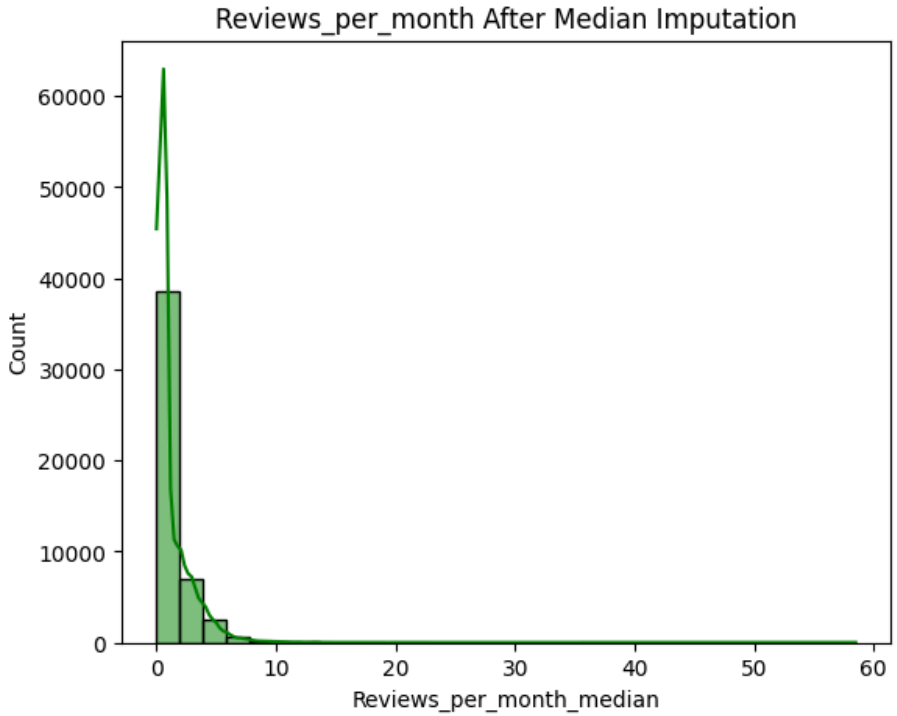

```
#Apply Mean & Median Imputation
mean_imputer = SimpleImputer(strategy='mean')
median_imputer = SimpleImputer(strategy='median')
```

```
4
5 df['Reviews_per_month_median'] = median_imputer.fit_transform(df[['Reviews_per_month']])
```

Handle Categorical Missing Values (Embarked)

```
1 plt.figure(figsize=(14,5))
2 plt.subplot(1,2,1)
3 sns.histplot(df['Reviews_per_month'], bins=30, kde=True, color="blue")
4 plt.title("Reviews_per_month Before Imputation")
5
6 plt.subplot(1,2,2)
7 sns.histplot(df['Reviews_per_month_median'], bins=30, kde=True, color="green")
8 plt.title("Reviews_per_month After Median Imputation")
9
10 plt.show()
11
12 mode_imputer = SimpleImputer(strategy='most_frequent')
13 # You can apply mode imputation to a categorical column if needed.
14 # For example, if you had a 'City' column with missing values, you could use:
15 # df['City'] = mode_imputer.fit_transform(df[['City']]).ravel()
```



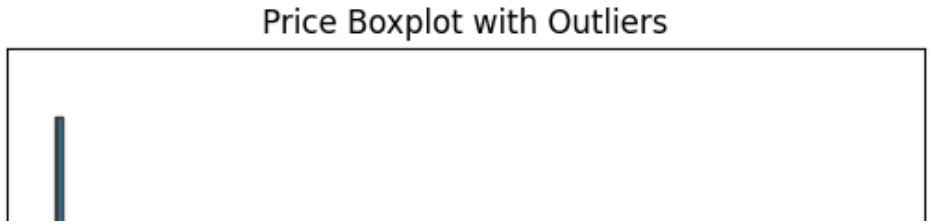



```
sns.boxplot(x=df['Price'])
plt.title("Price Boxplot with Outliers")
plt.show()

Q1 = df['Price'].quantile(0.25)
Q3 = df['Price'].quantile(0.75)
IQR = Q3 - Q1
upper_whisker = Q3 + 1.5*IQR

outliers = df[df['Price'] > upper_whisker]
print("Number of outliers in Price:", len(outliers))
df['Price_capped'] = df['Price'].clip(upper=upper_whisker)

sns.boxplot(x=df['Price_capped'])
plt.title("Price Boxplot After Capping Outliers")
plt.show()
```


Price Boxplot with Outliers