Importing Necessary Libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
```

reading dataset

In [2]:
```
data = pd.read_csv("datasets.csv", encoding_errors = 'ignore')
```

Initial Exploration

In [3]:
```
data.head(2)
```

Out[3]:

	id	name	host_id	host_name	neighbourhood_group	neighbou
0	1312228.0	Rental unit in Brooklyn · ★5.0 · 1 bedroom	7130382	Walter	Brooklyn	Clint
1	45277537.0	Rental unit in New York · ★4.67 · 2 bedrooms ·...	51501835	Jeniffer	Manhattan	Hell's K

2 rows × 22 columns

In [4]:
```
data.columns
```

Out[4]:
```
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', 'number_of_reviews_ltm', 'license', 'rating',
       'bedrooms', 'beds', 'baths'],
      dtype='object')
```

In [5]:
```
pd.set_option('display.float_format', '{:.2f}'.format)
```

In [6]:
```
data.describe()
```

Out[6]:

	id	host_id	latitude	longitude	price	m
count	20770.00	20770.00	20763.00	20763.00	20736.00	
mean	303385844987444096.00	174904903.02	40.73	-73.94	187.71	
std	390122084199058432.00	172565669.39	0.06	0.06	1023.25	
min	2595.00	1678.00	40.50	-74.25	10.00	
25%	27072602.75	20411843.75	40.68	-73.98	80.00	
50%	49928523.50	108699045.00	40.72	-73.95	125.00	
75%	722000000000000000.00	314399689.50	40.76	-73.92	199.00	
max	1050000000000000000.00	550403525.00	40.91	-73.71	100000.00	

In [7]:
```
data.tail(2)
```

Out[7]:

	id	name	host_id	host_name	neighbourhood_
20768	783000000000000000.00	Rental unit in New York · ★5.0 · 1 bedroom · 1...	163083101	Marissa	Mar
20769	566000000000000000.00	Rental unit in Queens · ★4.89 · 1 bedroom · 1 ...	93827372	Glenroy	C

2 rows × 22 columns

In [8]:
```
data.shape
```

Out[8]:
```
(20770, 22)
```

In [9]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20770 entries, 0 to 20769
Data columns (total 22 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   id                              20770 non-null  float64
 1   name                            20770 non-null  object 
 2   host_id                         20770 non-null  int64  
 3   host_name                       20770 non-null  object 
 4   neighbourhood_group             20770 non-null  object 
 5   neighbourhood                   20763 non-null  object 
 6   latitude                        20763 non-null  float64
 7   longitude                       20763 non-null  float64
 8   room_type                       20763 non-null  object 
 9   price                           20736 non-null  float64
 10  minimum_nights                  20763 non-null  float64
 11  number_of_reviews               20763 non-null  float64
 12  last_review                     20763 non-null  object 
 13  reviews_per_month               20763 non-null  float64
 14  calculated_host_listings_count  20763 non-null  float64
 15  availability_365                20763 non-null  float64
 16  number_of_reviews_ltm           20763 non-null  float64
 17  license                         20770 non-null  object 
 18  rating                          20770 non-null  object 
 19  bedrooms                        20770 non-null  object 
 20  beds                            20770 non-null  int64  
 21  baths                           20770 non-null  object 
dtypes: float64(10), int64(2), object(10)
memory usage: 3.5+ MB
```

In [ ]:

Data Cleaning

In [10]:
```
data.isnull().sum()


data.dropna(inplace = True)
```

In [11]:
```
data.isnull().sum()
```

Out[11]:
```
id                                0
name                              0
host_id                           0
host_name                         0
neighbourhood_group               0
neighbourhood                     0
latitude                          0
longitude                         0
room_type                         0
price                             0
minimum_nights                    0
number_of_reviews                 0
last_review                       0
reviews_per_month                 0
calculated_host_listings_count    0
availability_365                  0
number_of_reviews_ltm             0
license                           0
rating                            0
bedrooms                          0
beds                              0
baths                             0
dtype: int64
```

In [12]:
```
data.shape
```

Out[12]:
```
(20736, 22)
```

In [13]:
```
data.duplicated().sum()
```

Out[13]:
```
12
```

In [14]:
```
data[data.duplicated()]
```

Out[14]:

	id	name	host_id	host_name	neighbourhood
6	45277537.00	Rental unit in New York · ★4.67 · 2 bedrooms ·...	51501835	Jeniffer	Ma
7	971000000000000000.00	Rental unit in New York · ★4.17 · 1 bedroom · ...	528871354	Joshua	Ma
8	3857863.00	Rental unit in New York · ★4.64 · 1 bedroom · ...	19902271	John And Catherine	Ma
9	40896611.00	Condo in New York · ★4.91 · Studio · 1 bed · 1...	61391963	Stay With Vibe	Ma
10	49584983.00	Rental unit in New York · ★5.0 · 1 bedroom · 1...	51501835	Jeniffer	Ma
20736	799000000000000000.00	Rental unit in New York · 2 bedrooms · 2 beds ...	224733902	CozySuites Copake	Ma
20737	593000000000000000.00	Rental unit in New York · ★4.79 · 2 bedrooms ·...	23219783	Rob	Ma
20738	923000000000000000.00	Loft in New York · ★4.33 · 1 bedroom	520265731	Rodrigo	Ma

	id	name	host_id	host_name	neighbourhood
		· 2 beds ...			
20739	13361613.00	Rental unit in New York · ★4.89 · 2 bedrooms ·...	8961407	Jamie	Ma
20740	51195659.00	Rental unit in New York · Studio · 1 bed · 1 bath	51501835	Jeniffer	Ma
20741	25234732.00	Rental unit in New York · ★4.41 · 1 bedroom · ...	1497427	Mara	Ma
20742	3339399.00	Rental unit in New York · ★4.73 · 1 bedroom · ...	2119276	Urban Furnished	Ma

12 rows × 22 columns

In [15]:
```
data.drop_duplicates(inplace = True)
```

In [16]:
```
data.duplicated().sum()
```

Out[16]:
```
0
```

In [17]:
```
data[data.duplicated()]
```

Out[17]:

id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude

0 rows × 22 columns

In [18]:
```
data.dtypes
```

Out[18]:
```
id                                float64
name                               object
host_id                             int64
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                             float64
minimum_nights                    float64
number_of_reviews                 float64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count    float64
availability_365                  float64
number_of_reviews_ltm             float64
license                            object
rating                             object
bedrooms                           object
beds                                int64
baths                              object
dtype: object
```

In [19]:
```
data.id = data['id'].astype(object)
```

In [20]:
```
data.id.info()
```

```
<class 'pandas.core.series.Series'>
Index: 20724 entries, 0 to 20769
Series name: id
Non-Null Count  Dtype
--------------  -----
20724 non-null  object
dtypes: object(1)
memory usage: 323.8+ KB
```

In [21]:
```
data.dtypes
```

Out[21]:
```
id                                object
name                              object
host_id                            int64
host_name                         object
neighbourhood_group               object
neighbourhood                     object
latitude                         float64
longitude                        float64
room_type                         object
price                            float64
minimum_nights                   float64
number_of_reviews                float64
last_review                       object
reviews_per_month                float64
calculated_host_listings_count   float64
availability_365                 float64
number_of_reviews_ltm            float64
license                           object
rating                            object
bedrooms                          object
beds                               int64
baths                             object
dtype: object
```

In [22]:
```
data.host_id = data.host_id.astype(object)
```

In [23]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 20724 entries, 0 to 20769
Data columns (total 22 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   id                              20724 non-null  object 
 1   name                            20724 non-null  object 
 2   host_id                         20724 non-null  object 
 3   host_name                       20724 non-null  object 
 4   neighbourhood_group             20724 non-null  object 
 5   neighbourhood                   20724 non-null  object 
 6   latitude                        20724 non-null  float64
 7   longitude                       20724 non-null  float64
 8   room_type                       20724 non-null  object 
 9   price                           20724 non-null  float64
 10  minimum_nights                  20724 non-null  float64
 11  number_of_reviews               20724 non-null  float64
 12  last_review                     20724 non-null  object 
 13  reviews_per_month               20724 non-null  float64
 14  calculated_host_listings_count  20724 non-null  float64
 15  availability_365                20724 non-null  float64
 16  number_of_reviews_ltm           20724 non-null  float64
 17  license                         20724 non-null  object 
 18  rating                          20724 non-null  object 
 19  bedrooms                        20724 non-null  object 
 20  beds                            20724 non-null  int64  
 21  baths                           20724 non-null  object 
dtypes: float64(9), int64(1), object(12)
memory usage: 3.6+ MB
```

Data Analysis

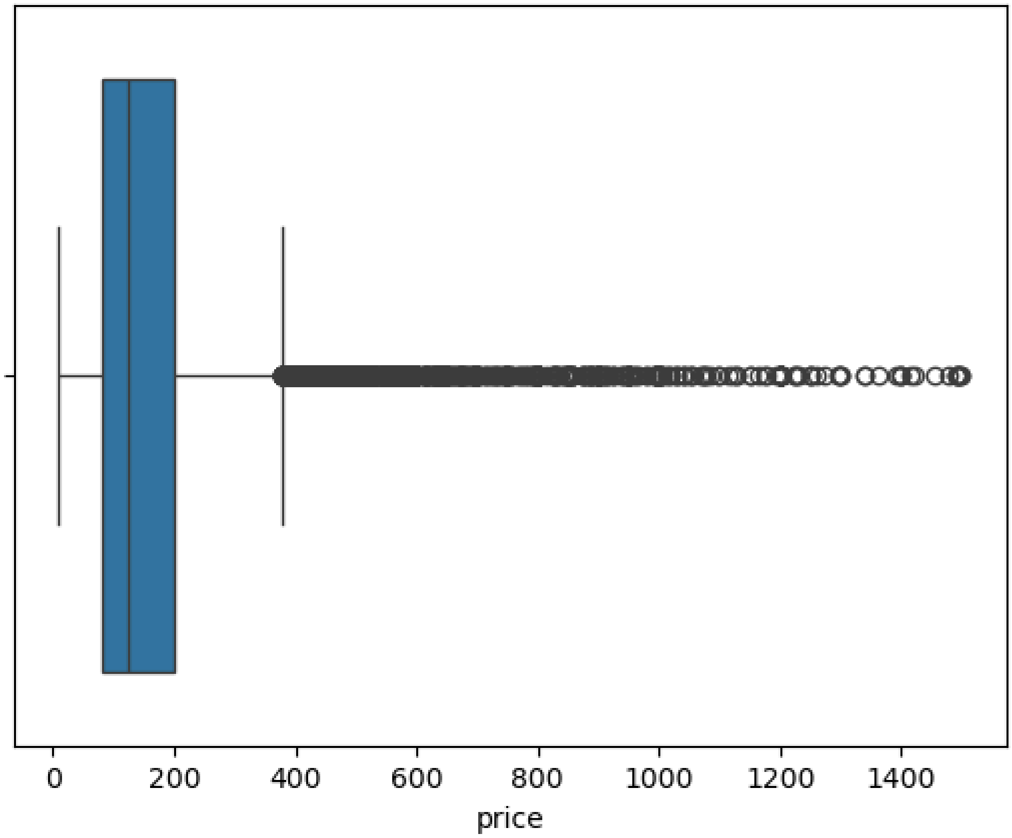
Univariate Analysis

In [25]:
```
# Price Distribution

data['price']
```

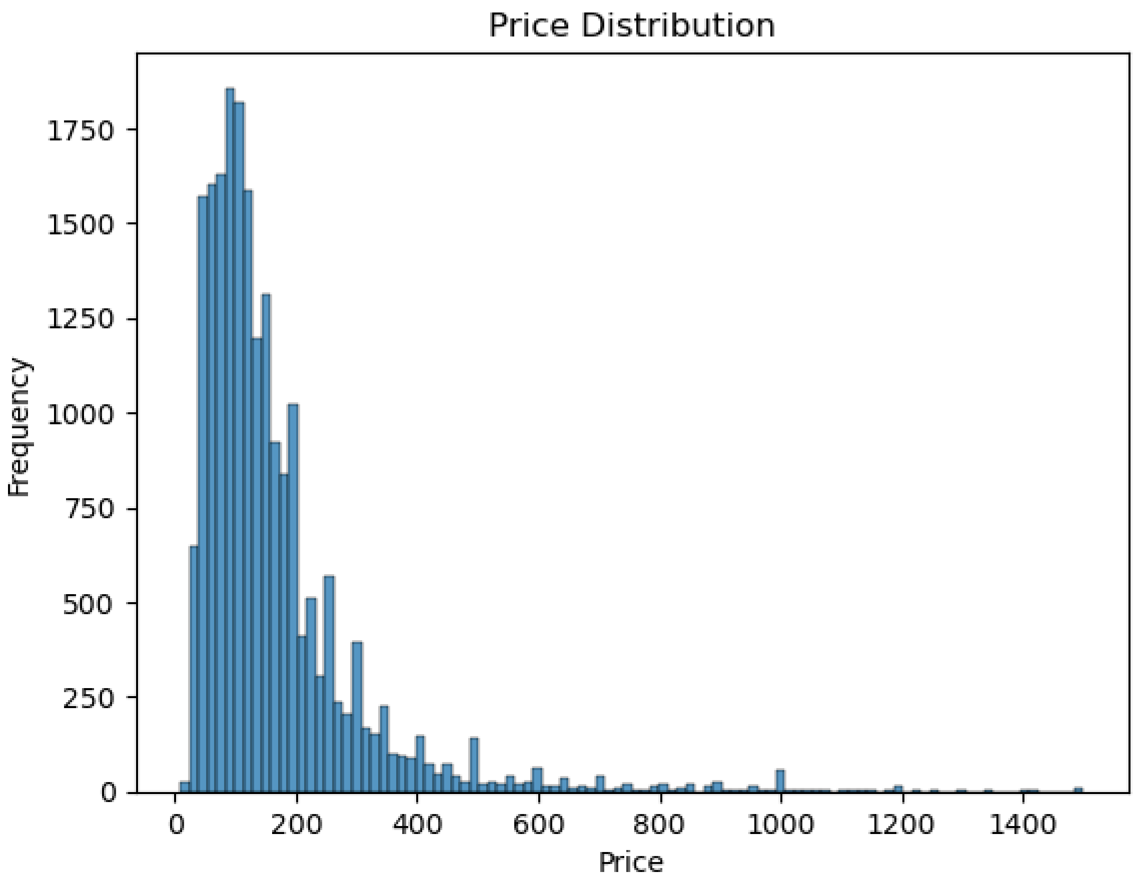
Out[25]:
```
0         55.00
1        144.00
2        187.00
3        120.00
4         85.00
          ...  
20765     45.00
20766    105.00
20767    299.00
20768    115.00
20769    102.00
Name: price, Length: 20724, dtype: float64
```

In [34]:
```python
# identifying outliers in price

df = data[data['price'] < 1500]
sns.boxplot(data = df, x = 'price')
```

Out[34]: <Axes: xlabel='price'>



In [39]:
```python
sns.histplot(data =df, bins = 100, x = 'price')
plt.title('Price Distribution')
plt.xlabel('Price')
plt.ylabel('Frequency')
plt.show()
```



Observation :

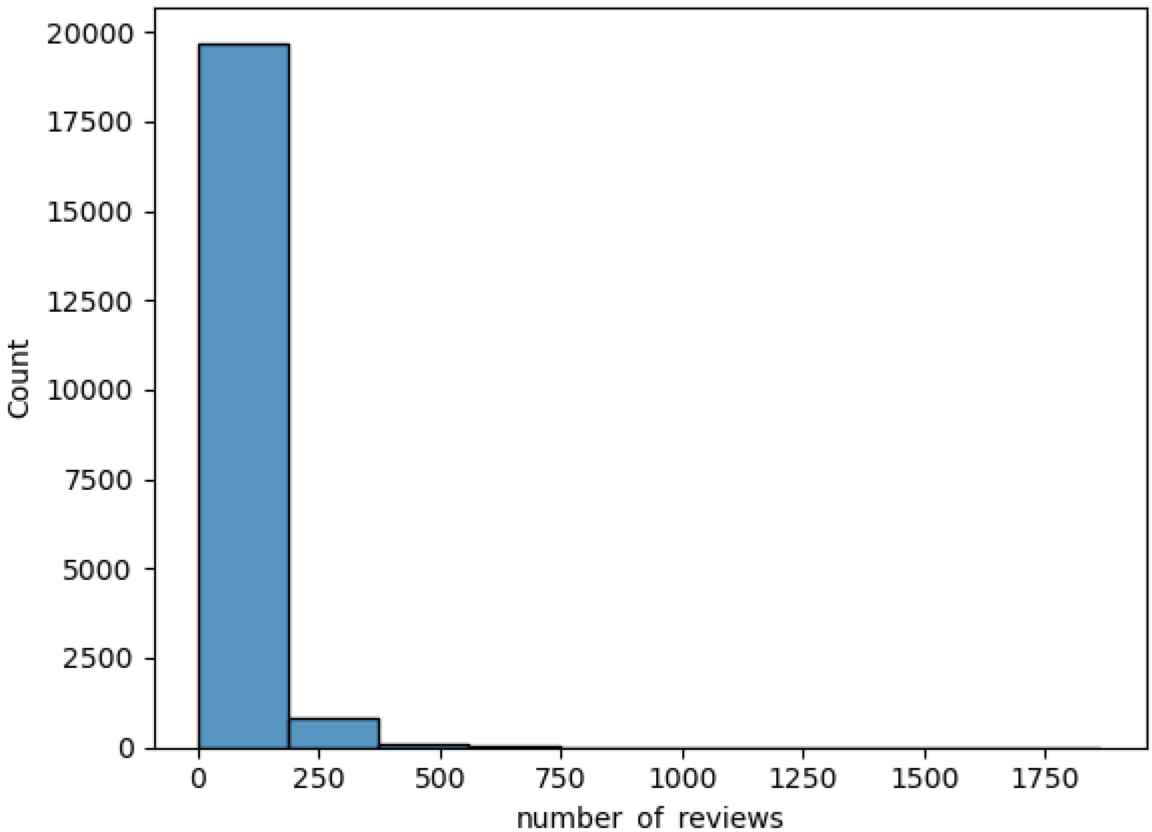
Most of the Price distribution from 10 to 400$

In [41]:
```
df.columns
```

Out[41]:
```
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', 'number_of_reviews_ltm', 'license', 'rating',
       'bedrooms', 'beds', 'baths'],
      dtype='object')
```

In [73]:
```
sns.histplot(data = df, x = 'number_of_reviews', bins = 10)
```

Out[73]:
```
<Axes: xlabel='number_of_reviews', ylabel='Count'>
```



Observation :

Most of the reviews is from 0 to 200

In [74]:
```
df.dtypes
```

Out[74]:
```
id                                 object
name                               object
host_id                            object
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                             float64
minimum_nights                    float64
number_of_reviews                 float64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count    float64
availability_365                  float64
number_of_reviews_ltm             float64
license                            object
rating                             object
bedrooms                           object
beds                                int64
baths                              object
dtype: object
```

In [80]:
```
df.head(2)
```

Out[80]:

	id	name	host_id	host_name	neighbourhood_group	neighbou
0	1312228.00	Rental unit in Brooklyn · ★5.0 · 1 bedroom	7130382	Walter	Brooklyn	Clir
1	45277537.00	Rental unit in New York · ★4.67 · 2 bedrooms ·...	51501835	Jeniffer	Manhattan	Hell's

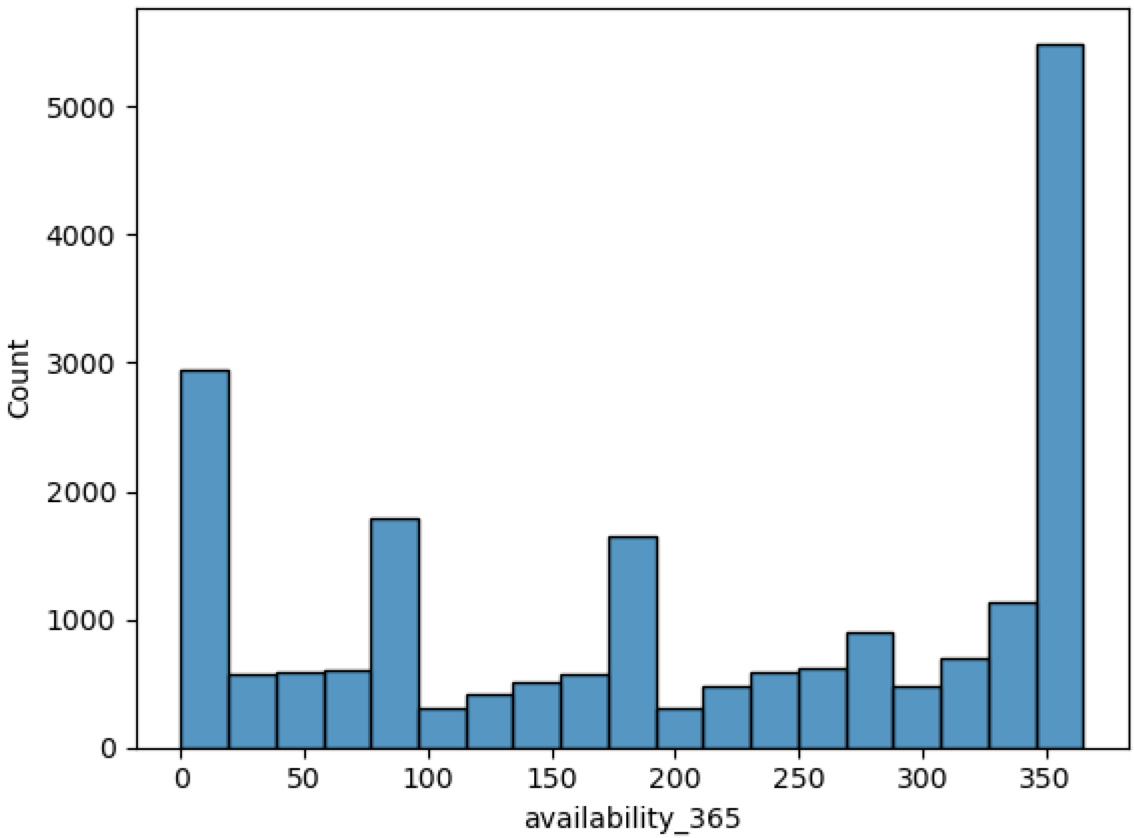
2 rows × 22 columns

In [79]:
```
df.availability_365
```

Out[79]:
```
0          0.00
1        364.00
2        343.00
3        363.00
4        335.00
          ...
20765    157.00
20766      0.00
20767      0.00
20768    363.00
20769      0.00
Name: availability_365, Length: 20636, dtype: float64
```

In [82]:
```
sns.histplot(data = df, x = 'availability_365')
```

Out[82]: <Axes: xlabel='availability_365', ylabel='Count'>



Observation :

Most of the Hotels & Rooms are available for 365 days

In [84]:
```
df.dtypes
```

Out[84]:
```
id                                 object
name                               object
host_id                            object
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                             float64
minimum_nights                    float64
number_of_reviews                 float64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count    float64
availability_365                  float64
number_of_reviews_ltm             float64
license                            object
rating                             object
bedrooms                           object
beds                                int64
baths                              object
dtype: object
```

In [90]:
```
avg_price_of_neighbour = df.groupby('neighbourhood_group')['price'].mean().sor
avg_price_of_neighbour
```

Out[90]:
```
neighbourhood_group
Manhattan        204.15
Brooklyn         155.14
Queens           121.68
Staten Island    118.78
Bronx            107.99
Name: price, dtype: float64
```

Feature Engineering

In [ ]:
```
df.groupby('number_of_reviews')['price'].max().reset_index()
```

Out[ ]:

	number_of_reviews	price
0	1.00	1456.00
1	2.00	1495.00
2	3.00	1364.00
3	4.00	1029.00
4	5.00	1495.00
...	...	...
462	1188.00	161.00
463	1201.00	177.00
464	1574.00	148.00
465	1618.00	163.00
466	1865.00	144.00

467 rows × 2 columns

In [93]:

```
df.dtypes
```

Out[93]:

```
id                                 object
name                               object
host_id                            object
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                             float64
minimum_nights                    float64
number_of_reviews                 float64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count    float64
availability_365                  float64
number_of_reviews_ltm             float64
license                            object
rating                             object
bedrooms                           object
beds                                int64
baths                              object
dtype: object
```

In [98]:

```
df['price per bed'] = df['price']/df['beds']
df['price per bed']
df.head(2)
```

Out[98]:

	id	name	host_id	host_name	neighbourhood_group	neighbou
0	1312228.00	Rental unit in Brooklyn · ★5.0 · 1 bedroom	7130382	Walter	Brooklyn	Clir
1	45277537.00	Rental unit in New York · ★4.67 · 2 bedrooms ·...	51501835	Jeniffer	Manhattan	Hell's

2 rows × 23 columns

In [103…

```
avg_price_per_bed = df.groupby('neighbourhood_group')['price per bed'].mean().
avg_price_per_bed
```

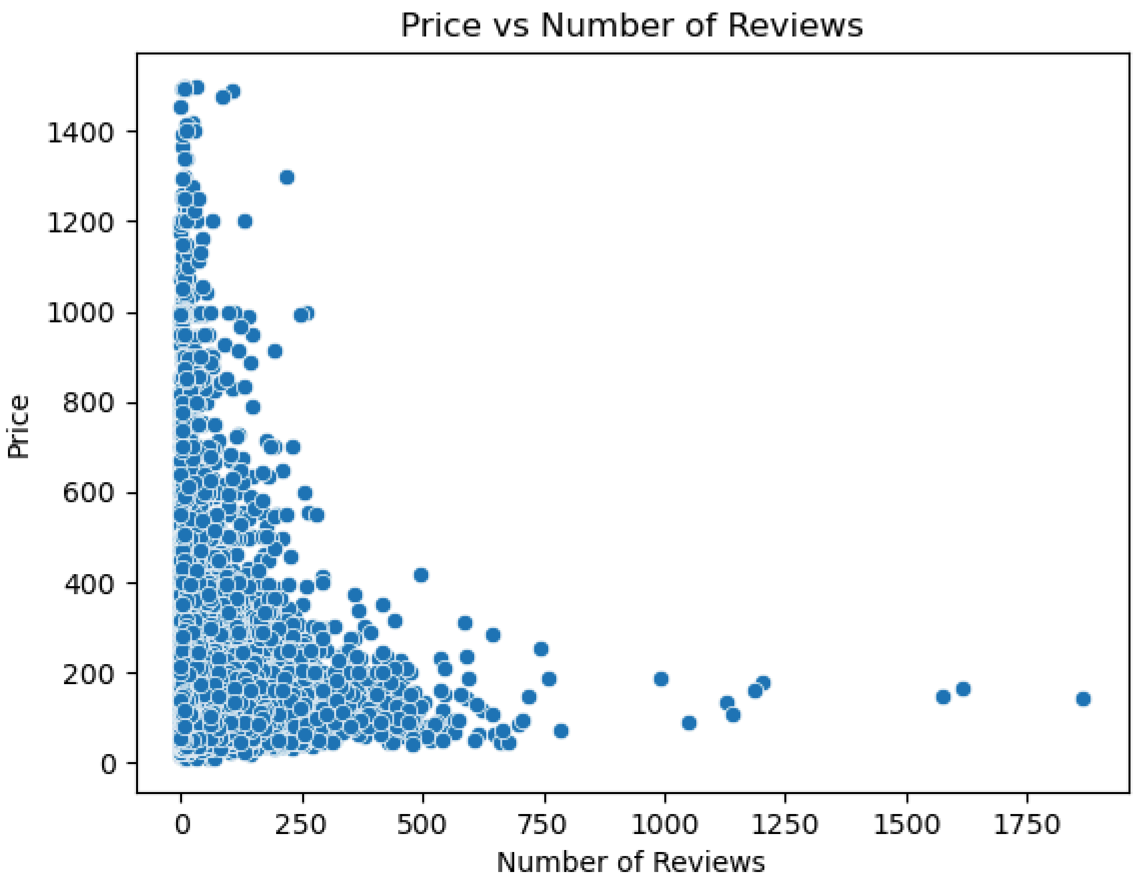
Out[103…

	neighbourhood_group	price per bed
0	Bronx	74.71
1	Brooklyn	99.79
2	Manhattan	138.71
3	Queens	76.34
4	Staten Island	67.73

Bi Variate Analysis

In [48]:

```
sns.scatterplot(data = df, x = 'number_of_reviews', y = 'price')
plt.title('Price vs Number of Reviews')
plt.xlabel('Number of Reviews')
plt.ylabel('Price')
plt.show()
```



Observation :

As the number of reviews increase the price is decreases they both are in inversely proportional relationship

In [55]:
```
df.columns
```

Out[55]:
```
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', 'number_of_reviews_ltm', 'license', 'rating',
       'bedrooms', 'beds', 'baths'],
      dtype='object')
```

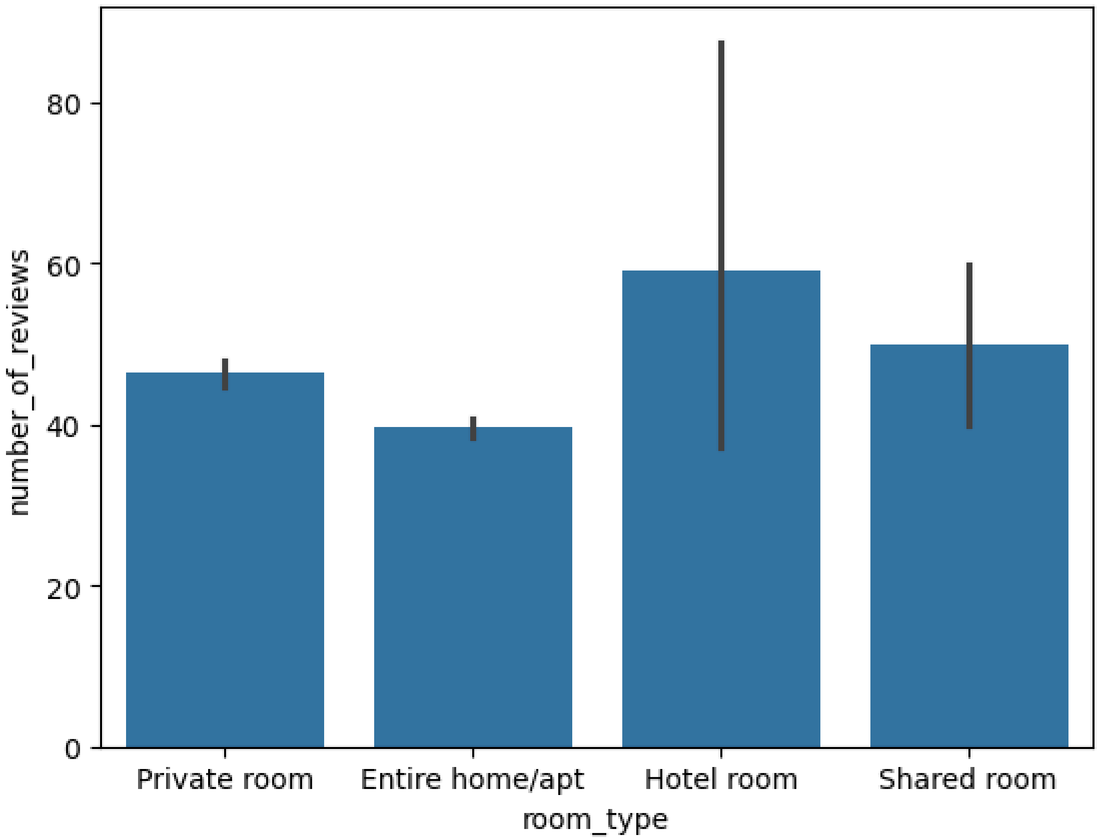
In [68]:
```
df.groupby('room_type')['number_of_reviews'].max().reset_index()
```

Out[68]:

	room_type	number_of_reviews
0	Entire home/apt	1139.00
1	Hotel room	745.00
2	Private room	1865.00
3	Shared room	506.00

In [69]:
```
sns.barplot(data = df, x = 'room_type', y = 'number_of_reviews')
```

Out[69]: `<Axes: xlabel='room_type', ylabel='number_of_reviews'>`

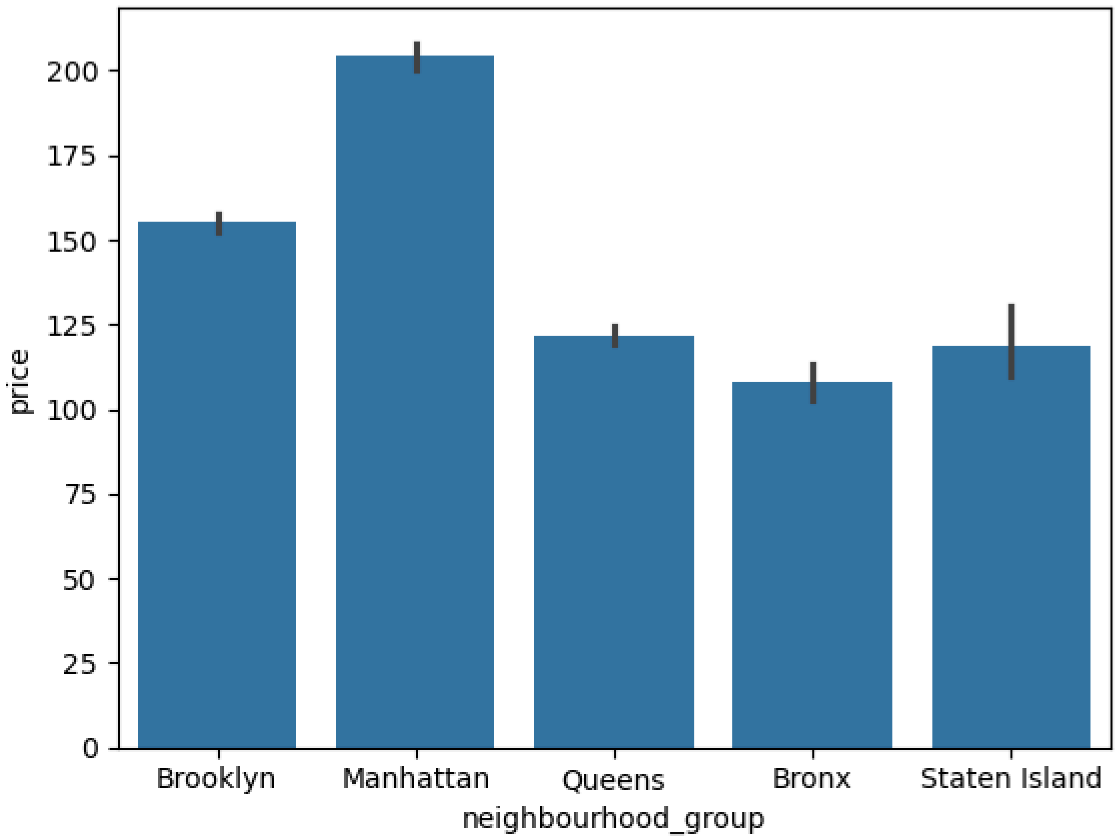


Observation :

- *As we see the Hotel Room has the max number of reviews*

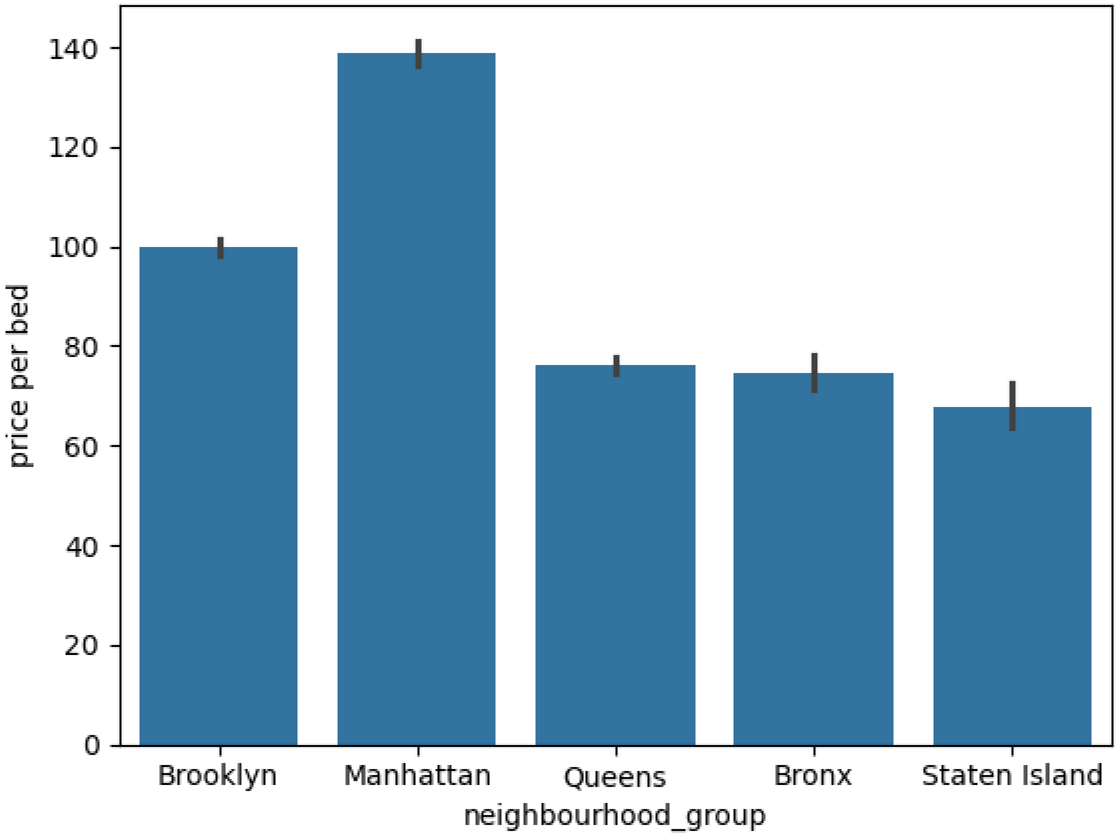
In [ ]:
```
sns.barplot(data =df, x='neighbourhood_group', y='price')
```

Out[ ]: `<Axes: xlabel='neighbourhood_group', ylabel='price'>`



In [106…
```
sns.barplot(data =df, x='neighbourhood_group', y='price per bed')
```

Out[106…
```
<Axes: xlabel='neighbourhood_group', ylabel='price per bed'>
```

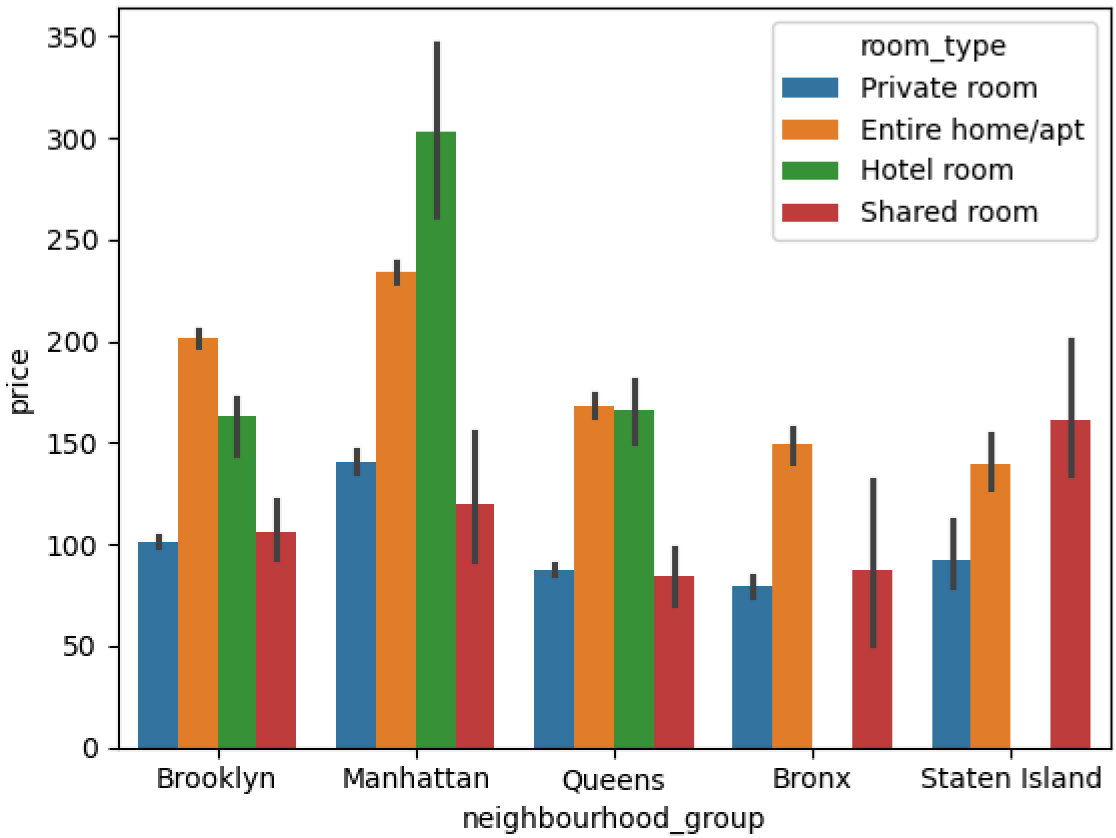


Observation :

- *As we see Manhattan neighbourhood have the highest price per bed*

In [108…
```
sns.barplot(data =df, x='neighbourhood_group', y='price', hue = 'room_type')
```

Out[108…
```
<Axes: xlabel='neighbourhood_group', ylabel='price'>
```



Observation :

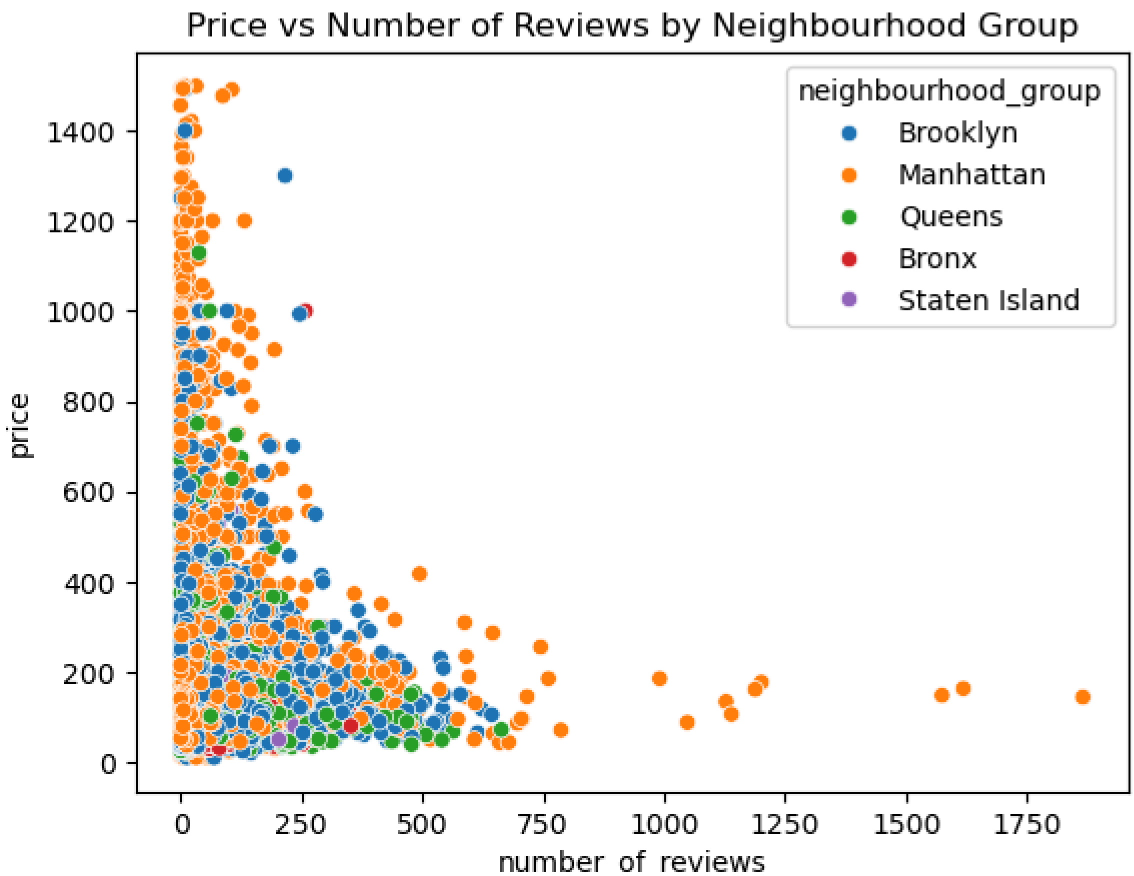
- *Staten Island shared room price more then Manhattan shared room price*

In [114…

```
sns.scatterplot(data = df, x ='number_of_reviews', y ='price', hue = 'neighbou
plt.title('Price vs Number of Reviews by Neighbourhood Group')
```

Out[114…

```
Text(0.5, 1.0, 'Price vs Number of Reviews by Neighbourhood Group')
```



Observation :

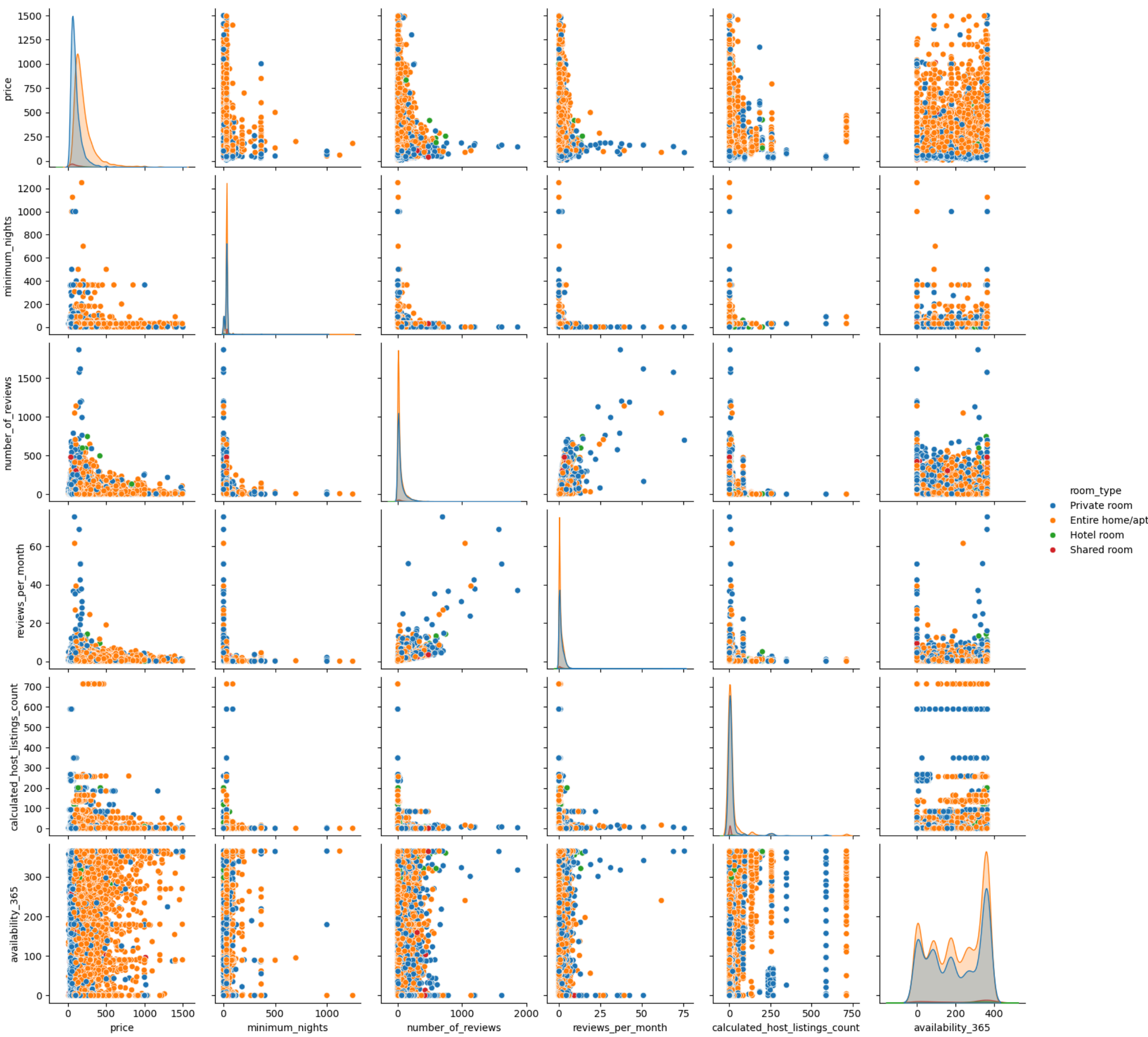
- *As we see number of reviews of decreases the price is increasing*

In [115…

```
df.dtypes
```

Out[115…
```
id                                 object
name                               object
host_id                            object
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                             float64
minimum_nights                    float64
number_of_reviews                 float64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count    float64
availability_365                  float64
number_of_reviews_ltm             float64
license                            object
rating                             object
bedrooms                           object
beds                                int64
baths                              object
price per bed                     float64
dtype: object
```

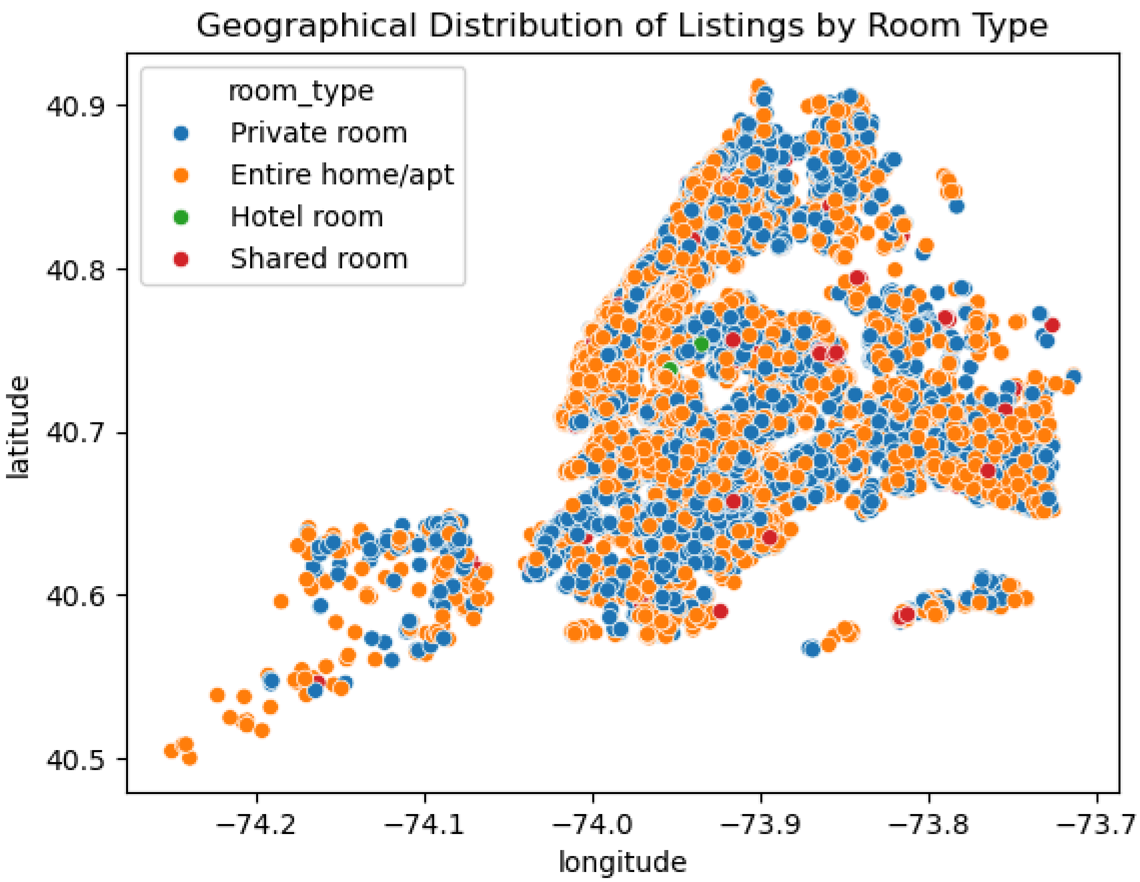
In [129…
```
sns.pairplot(data = df, vars=df[['price', 'minimum_nights', 'number_of_reviews
plt.show()
```



Geographical Distribution

In [133…

```
sns.scatterplot(data = df, x = 'longitude', y = 'latitude', hue = 'room_type')
plt.title('Geographical Distribution of Listings by Room Type')
plt.show()
```



Observation :

- Hotel room are less as compared to another room types

In [134…

```
df.dtypes
```

Out[134…
```
id                                object
name                              object
host_id                           object
host_name                         object
neighbourhood_group               object
neighbourhood                     object
latitude                         float64
longitude                        float64
room_type                         object
price                            float64
minimum_nights                   float64
number_of_reviews                float64
last_review                       object
reviews_per_month                float64
calculated_host_listings_count   float64
availability_365                 float64
number_of_reviews_ltm            float64
license                           object
rating                            object
bedrooms                          object
beds                               int64
baths                             object
price per bed                    float64
dtype: object
```

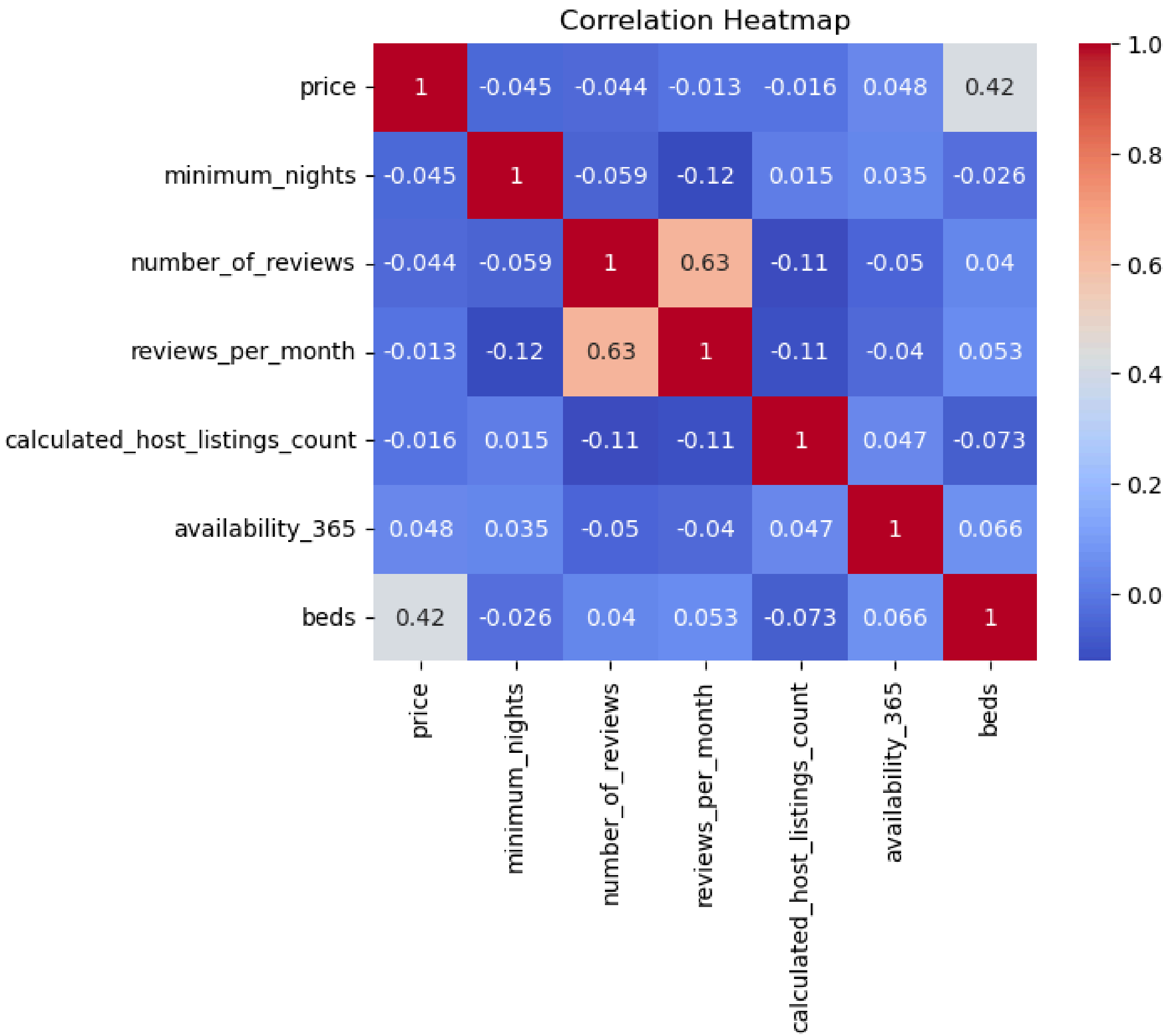
In [ ]:

In [145…
```
corr = df[['price', 'minimum_nights', 'number_of_reviews', 'reviews_per_month'
corr
```

Out[145…

	price	minimum_nights	number_of_reviews	revie
price	1.00	-0.04	-0.04	
minimum_nights	-0.04	1.00	-0.06	
number_of_reviews	-0.04	-0.06	1.00	
reviews_per_month	-0.01	-0.12	0.63	
calculated_host_listings_count	-0.02	0.01	-0.11	
availability_365	0.05	0.04	-0.05	
beds	0.42	-0.03	0.04	

In [146…
```
sns.heatmap(data =corr, annot = True, cmap = 'coolwarm' )
plt.title('Correlation Heatmap')
plt.figure(figsize=(6,6))
plt.show()
```




```
<Figure size 600x600 with 0 Axes>
```

Observation :

- *As the price increases the number of beds also increases*

```
In [ ]:
```