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

In [ ]:
```
df = pd.read_csv("airbnbdata.csv")
```

```
C:\Users\vikas\AppData\Local\Temp\ipykernel_15248\3860532478.py:1: DtypeWarning: 
Columns (25) have mixed types. Specify dtype option on import or set low_memory=F
alse.
  df = pd.read_csv("airbnbdata.csv")
```

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

Out[ ]:

	id	NAME	host id	host_identity_verified	host name	neighbourhoo grou
0	1001254	Clean & quiet apt home by the park	80014485718	unconfirmed	Madaline	Brookl
1	1002102	Skylit Midtown Castle	52335172823	verified	Jenna	Manhatta
2	1002403	THE VILLAGE OF HARLEM....NEW YORK !	78829239556	NaN	Elise	Manhatta
3	1002755	NaN	85098326012	unconfirmed	Garry	Brookl
4	1003689	Entire Apt: Spacious Studio/Loft by central park	92037596077	verified	Lyndon	Manhatta

5 rows × 26 columns

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

Out[ ]:
```
Index(['id', 'NAME', 'host id', 'host_identity_verified', 'host name',
       'neighbourhood group', 'neighbourhood', 'lat', 'long', 'country',
       'country code', 'instant_bookable', 'cancellation_policy', 'room type',
       'Construction year', 'price', 'service fee', 'minimum nights',
       'number of reviews', 'last review', 'reviews per month',
       'review rate number', 'calculated host listings count',
       'availability 365', 'house_rules', 'license'],
      dtype='object')
```

Checking Missing Values

In [ ]:
```
df.isnull().sum()
```

Out[ ]:
```
id                                  0
NAME                              250
host id                             0
host_identity_verified            289
host name                         406
neighbourhood group                29
neighbourhood                      16
lat                                 8
long                                8
country                           532
country code                      131
instant_bookable                  105
cancellation_policy                76
room type                           0
Construction year                 214
price                             247
service fee                       273
minimum nights                    409
number of reviews                 183
last review                     15893
reviews per month               15879
review rate number                326
calculated host listings count    319
availability 365                  448
house_rules                     52131
license                        102597
dtype: int64
```

In [ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 102599 entries, 0 to 102598
Data columns (total 26 columns):
 #   Column                          Non-Null Count   Dtype  
---  ------                          --------------   -----  
 0   id                              102599 non-null  int64  
 1   NAME                            102349 non-null  object 
 2   host id                         102599 non-null  int64  
 3   host_identity_verified          102310 non-null  object 
 4   host name                       102193 non-null  object 
 5   neighbourhood group             102570 non-null  object 
 6   neighbourhood                   102583 non-null  object 
 7   lat                             102591 non-null  float64
 8   long                            102591 non-null  float64
 9   country                         102067 non-null  object 
 10  country code                    102468 non-null  object 
 11  instant_bookable                102494 non-null  object 
 12  cancellation_policy             102523 non-null  object 
 13  room type                       102599 non-null  object 
 14  Construction year               102385 non-null  float64
 15  price                           102352 non-null  object 
 16  service fee                     102326 non-null  object 
 17  minimum nights                  102190 non-null  float64
 18  number of reviews               102416 non-null  float64
 19  last review                     86706 non-null   object 
 20  reviews per month               86720 non-null   float64
 21  review rate number              102273 non-null  float64
 22  calculated host listings count  102280 non-null  float64
 23  availability 365                102151 non-null  float64
 24  house_rules                     50468 non-null   object 
 25  license                         2 non-null       object 
dtypes: float64(9), int64(2), object(15)
memory usage: 20.4+ MB
```

Handling Missing Values in the Dataset

In [ ]:
```
df['last review'] = pd.to_datetime(df['last review'], errors = 'coerce')
```

In [ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 102599 entries, 0 to 102598
Data columns (total 26 columns):
 #   Column                          Non-Null Count   Dtype         
---  ------                          --------------   -----         
 0   id                              102599 non-null  int64         
 1   NAME                            102349 non-null  object        
 2   host id                         102599 non-null  int64         
 3   host_identity_verified          102310 non-null  object        
 4   host name                       102193 non-null  object        
 5   neighbourhood group             102570 non-null  object        
 6   neighbourhood                   102583 non-null  object        
 7   lat                             102591 non-null  float64       
 8   long                            102591 non-null  float64       
 9   country                         102067 non-null  object        
 10  country code                    102468 non-null  object        
 11  instant_bookable                102494 non-null  object        
 12  cancellation_policy             102523 non-null  object        
 13  room type                       102599 non-null  object        
 14  Construction year               102385 non-null  float64       
 15  price                           102352 non-null  object        
 16  service fee                     102326 non-null  object        
 17  minimum nights                  102190 non-null  float64       
 18  number of reviews               102416 non-null  float64       
 19  last review                     86706 non-null   datetime64[ns]
 20  reviews per month               86720 non-null   float64       
 21  review rate number              102273 non-null  float64       
 22  calculated host listings count  102280 non-null  float64       
 23  availability 365                102151 non-null  float64       
 24  house_rules                     50468 non-null   object        
 25  license                         2 non-null       object        
dtypes: datetime64[ns](1), float64(9), int64(2), object(14)
memory usage: 20.4+ MB
```

In [ ]:
```
df.fillna({'reviews per month': 0, 'last review': df['last review'].min()}, inpl
```

In [ ]:
```
df.dropna(subset = ['NAME', 'host name'], inplace = True)
```

In [ ]:
```
df.isnull().sum()
```

Out[ ]:
```
id                                  0
NAME                                0
host id                             0
host_identity_verified            276
host name                           0
neighbourhood group                26
neighbourhood                      16
lat                                 8
long                                8
country                           526
country code                      122
instant_bookable                   96
cancellation_policy                70
room type                           0
Construction year                 200
price                             239
service fee                       268
minimum nights                    403
number of reviews                 182
last review                         0
reviews per month                   0
review rate number                314
calculated host listings count    318
availability 365                  420
house_rules                     51867
license                        101947
dtype: int64
```

In [ ]:
```
df = df.drop(columns =['license', 'house_rules'], errors = 'ignore')
```

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

Out[ ]:

	id	NAME	host id	host_identity_verified	host name	neighbourhood group
0	1001254	Clean & quiet apt home by the park	80014485718	unconfirmed	Madaline	Brookl
1	1002102	Skylit Midtown Castle	52335172823	verified	Jenna	Manhatta
2	1002403	THE VILLAGE OF HARLEM....NEW YORK !	78829239556	NaN	Elise	Manhatta
4	1003689	Entire Apt: Spacious Studio/Loft by central park	92037596077	verified	Lyndon	Manhatta
5	1004098	Large Cozy 1 BR Apartment In Midtown East	45498551794	verified	Michelle	Manhatta

5 rows × 24 columns

In [ ]:
```
# Remove dollar signs and Convert to Float
df['price'] = df['price'].replace('[\$,]', '', regex=True).astype(float)
df['service fee'] = df['service fee'].replace('[\$,]', '', regex=True).astype(fl
```

```
<>:2: SyntaxWarning: invalid escape sequence '\$'
<>:3: SyntaxWarning: invalid escape sequence '\$'
<>:2: SyntaxWarning: invalid escape sequence '\$'
<>:3: SyntaxWarning: invalid escape sequence '\$'
C:\Users\vikas\AppData\Local\Temp\ipykernel_15248\4032228593.py:2: SyntaxWarning:
invalid escape sequence '\$'
  df['price'] = df['price'].replace('[\$,]', '', regex=True).astype(float)
C:\Users\vikas\AppData\Local\Temp\ipykernel_15248\4032228593.py:3: SyntaxWarning:
invalid escape sequence '\$'
  df['service fee'] = df['service fee'].replace('[\$,]', '', regex=True).astype(f
loat)
```

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

Out[ ]:

	id	NAME	host id	host_identity_verified	host name	neighbourhood group
0	1001254	Clean & quiet apt home by the park	80014485718	unconfirmed	Madaline	Brookl
1	1002102	Skylit Midtown Castle	52335172823	verified	Jenna	Manhatta
2	1002403	THE VILLAGE OF HARLEM....NEW YORK !	78829239556	NaN	Elise	Manhatta
4	1003689	Entire Apt: Spacious Studio/Loft by central park	92037596077	verified	Lyndon	Manhatta
5	1004098	Large Cozy 1 BR Apartment In Midtown East	45498551794	verified	Michelle	Manhatta

5 rows × 24 columns

Removing Duplicates

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

In [ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 101410 entries, 0 to 102057
Data columns (total 24 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   id                              101410 non-null  int64
 1   NAME                            101410 non-null  object
 2   host id                         101410 non-null  int64
 3   host_identity_verified          101134 non-null  object
 4   host name                       101410 non-null  object
 5   neighbourhood group             101384 non-null  object
 6   neighbourhood                   101394 non-null  object
 7   lat                             101402 non-null  float64
 8   long                            101402 non-null  float64
 9   country                         100884 non-null  object
 10  country code                    101288 non-null  object
 11  instant_bookable                101314 non-null  object
 12  cancellation_policy             101340 non-null  object
 13  room type                       101410 non-null  object
 14  Construction year               101210 non-null  float64
 15  price                           101171 non-null  float64
 16  service fee                     101142 non-null  float64
 17  minimum nights                  101016 non-null  float64
 18  number of reviews               101228 non-null  float64
 19  last review                     101410 non-null  datetime64[ns]
 20  reviews per month               101410 non-null  float64
 21  review rate number              101103 non-null  float64
 22  calculated host listings count  101092 non-null  float64
 23  availability 365                100990 non-null  float64
dtypes: datetime64[ns](1), float64(11), int64(2), object(10)
memory usage: 19.3+ MB
```

Descriptive Statistics

In [ ]:
```
df.describe()
```

Out[ ]:

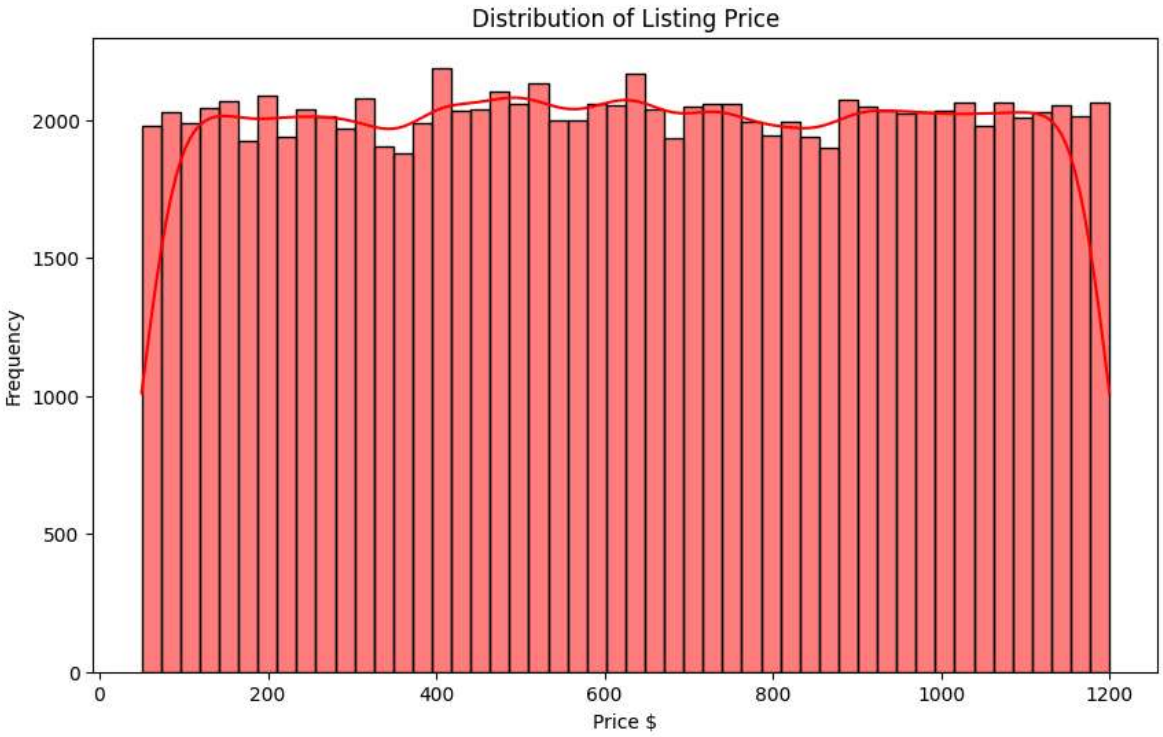
	id	host id	lat	long	Construction year	
count	1.014100e+05	1.014100e+05	101402.000000	101402.000000	101210.000000	10117
mean	2.920959e+07	4.926155e+10	40.728082	-73.949663	2012.486908	62
min	1.001254e+06	1.236005e+08	40.499790	-74.249840	2003.000000	5
25%	1.507574e+07	2.459183e+10	40.688730	-73.982570	2007.000000	34
50%	2.922911e+07	4.912069e+10	40.722300	-73.954440	2012.000000	62
75%	4.328308e+07	7.399747e+10	40.762750	-73.932340	2017.000000	91
max	5.736742e+07	9.876313e+10	40.916970	-73.705220	2022.000000	120
std	1.626820e+07	2.853703e+10	0.055850	0.049474	5.765130	33

Visualization

What is the distribution of the listing prices ?

In [ ]:

```
plt.figure(figsize=(10,6))
sns.histplot(df['price'], bins = 50, kde = True, color = 'red')
plt.title('Distribution of Listing Price')
plt.xlabel('Price $')
plt.ylabel('Frequency')
plt.show()
```



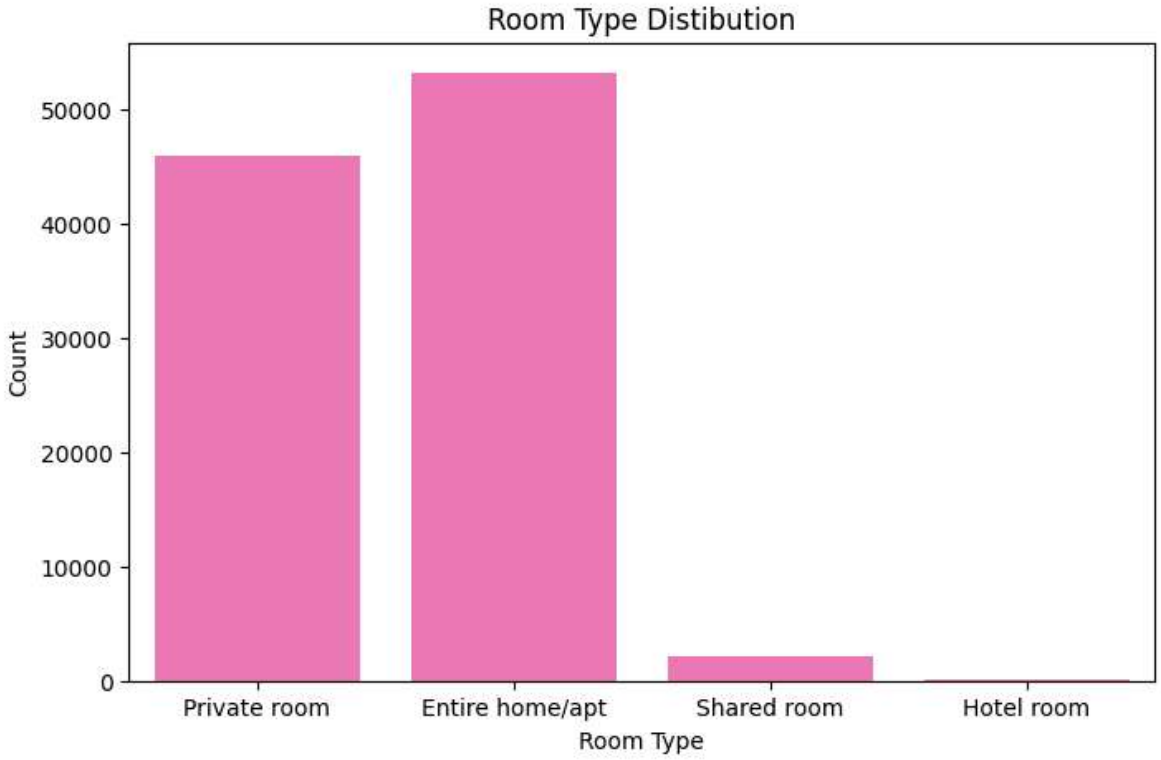
- The histogram shows a fairly even distribution of listing prices across different price ranges, indicating no particular concentration of listings in any specific price range.
- The KDE line helps visualize this even spread more clearly, confirming that the dataset contains listings with a wide variety of prices.

How are different room types are distributed ?

In [ ]:
```
df['room type']
```

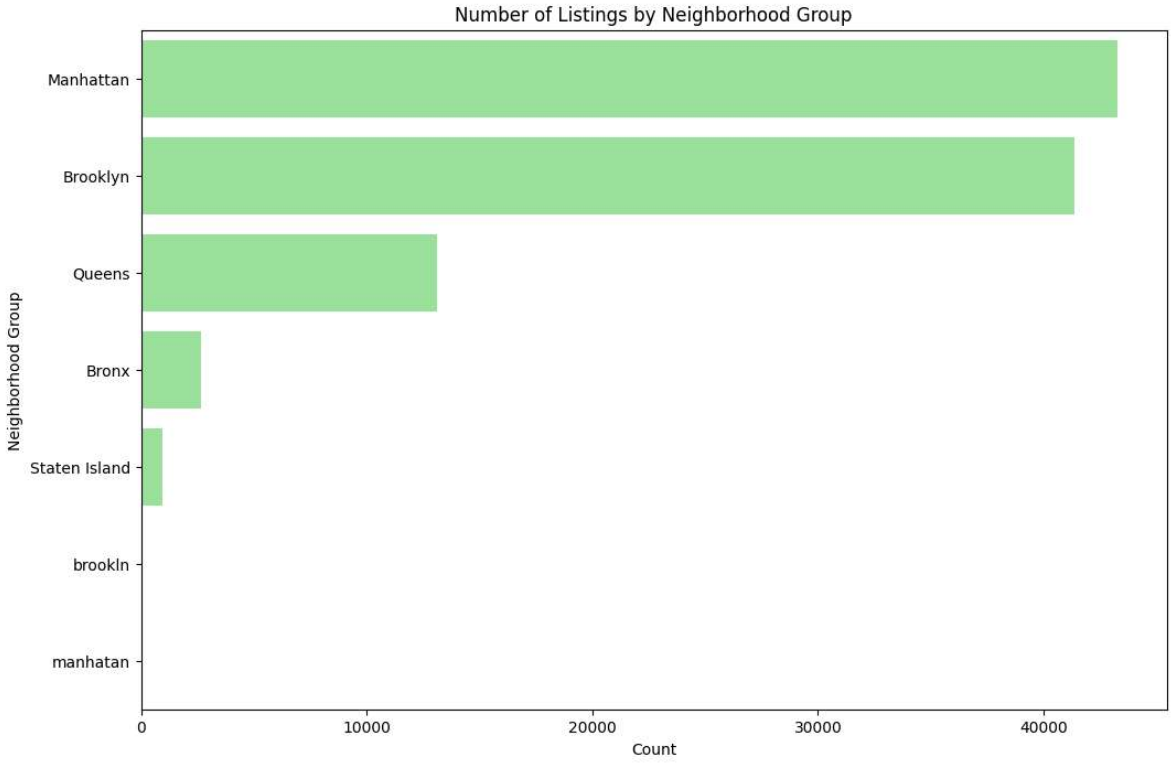
Out[ ]:
```
0            Private room
1         Entire home/apt
2            Private room
4         Entire home/apt
5         Entire home/apt
                 ...
102053       Private room
102054       Private room
102055    Entire home/apt
102056       Private room
102057    Entire home/apt
Name: room type, Length: 101410, dtype: object
```

In [ ]:
```
plt.figure(figsize= (8,5))
sns.countplot(x = 'room type', data = df, color = 'hotpink')
plt.title('Room Type Distibution')
plt.xlabel('Room Type')
plt.ylabel('Count')
plt.show()
```



How are listings distributed across different neighborhoods ?

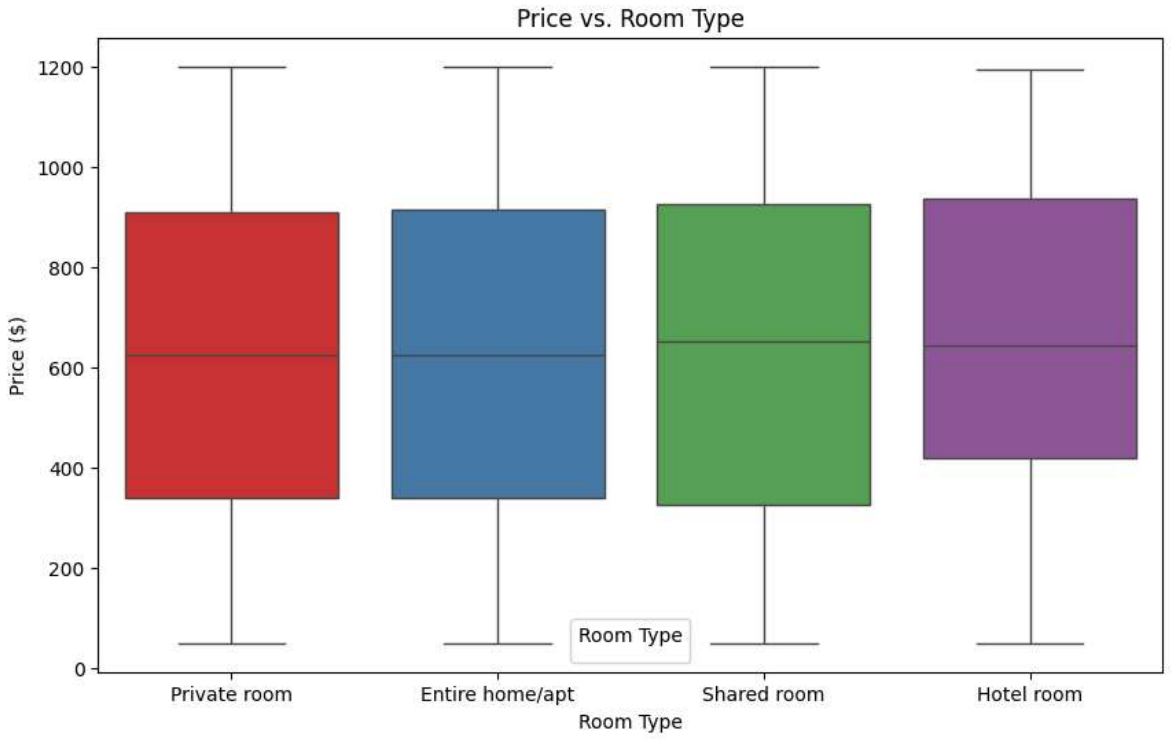
In [ ]:
```
plt.figure(figsize= (12,8))
sns.countplot(y = 'neighbourhood group', data = df, color = 'lightgreen', order
plt.title('Number of Listings by Neighborhood Group')
plt.xlabel('Count')
plt.ylabel('Neighborhood Group')
plt.show()
```



What is the relationship between price and room type

In [ ]:
```
plt.figure(figsize= (10, 6))
sns.boxplot(x = 'room type', y = 'price', hue = 'room type', data = df, palette
plt.title('Price vs. Room Type')
plt.xlabel('Room Type')
plt.ylabel('Price ($)')
plt.legend(title = 'Room Type')
plt.show()
```

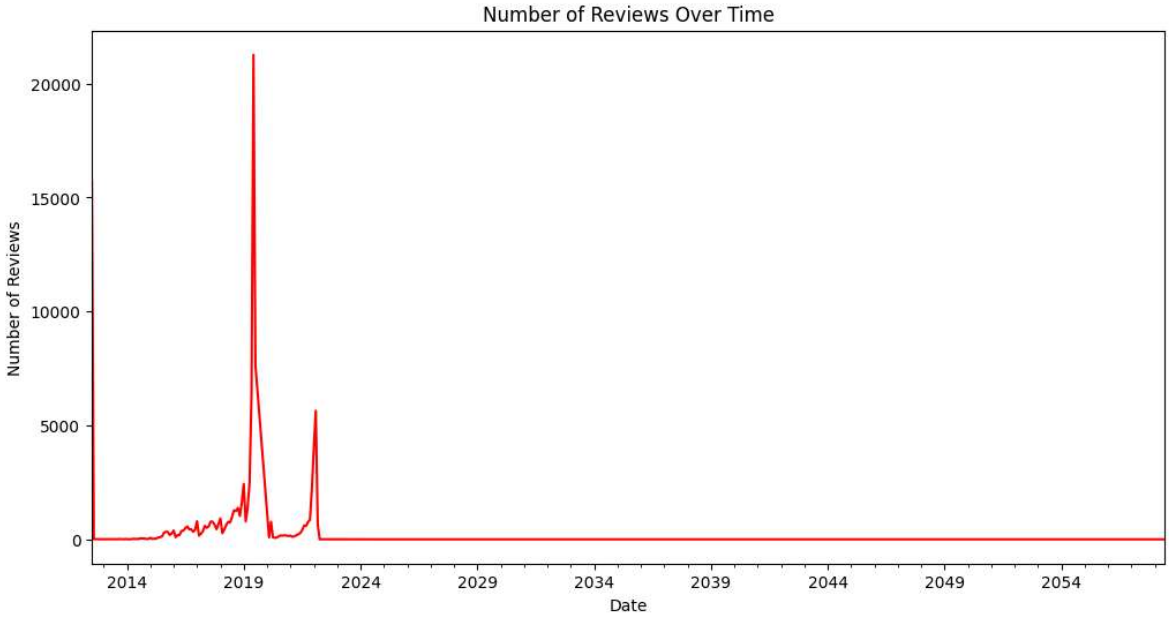
```
C:\Users\vikas\AppData\Local\Temp\ipykernel_15248\248839360.py:6: UserWarning: No
artists with labels found to put in legend.  Note that artists whose label start
with an underscore are ignored when legend() is called with no argument.
  plt.legend(title = 'Room Type')
```



How has the number of reviews has changed over time ?

In [ ]:
```
df['last review'] = pd.to_datetime(df['last review'])
reviews_over_time = df.groupby(df['last review'].dt.to_period('M')).size()

plt.figure(figsize= (12,6))
reviews_over_time.plot(kind = 'line', color = 'red')
plt.title('Number of Reviews Over Time')
plt.xlabel('Date')
plt.ylabel('Number of Reviews')
plt.show()
```


Number of Reviews Over Time
Number of Reviews
20000
15000
10000
5000
0
2014
2019
2024
2029
2034
2039
2044
2049
2054
Date