DSC-41 DA/BI

Methodology

airbnb

- CH VENKATA KAILASH
- VENKATA SIVA SAI TEJA NARAKULA
- CHAITALI DESAI

Airbnb Case Study

Objective

To prepare for the next best steps that Airbnb needs to take as a business, we have been asked to analyze a dataset consisting of various Airbnb listings in New York.

Problem Statement

For the past few months, Airbnb has seen a major decline in revenue. Now that the restrictions have started lifting and people have started to travel more, Airbnb wants to make sure that it is fully prepared for this change.

Tools Used

For the analysis, we have used the following tools

- Python Jupiter Notebook
- Tableau
- Microsoft Excel

Mainly to perform Data Cleaning and Data Analysis to come up with useful insights and business recommendations.

Derived/Calculated fields in Tableau

- Price bucket: for categorization of property in a structured manner
- No. of Night Group: for categorization based upon the number of minimum nights required for booking
- Revenue per stay: to check the revenue generated through each booking by multiplying min. of nights by the price

Let's dive into the details and approach we have used in a step-wise manner:

1. **Importing the data into Pandas Data Frame**

Importing usefull libraries

```
In [1]: import warnings
        warnings.filterwarnings('ignore')
```

```
In [2]: ## Importing usefull libraries

        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns
```

```
In [3]: ## To display all column and rows

        pd.set_option('display.max_columns', None)
        pd.set_option('display.max_rows', None)
        pd.set_option('display.width',None)
```

Importing and Reading the Dataset:

Out[4]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revie
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1	
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1	
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3	
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	1	
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10	
5	5099	Large Cozy 1 BR Apartment In	7322	Chris	Manhattan	Murray Hill	40.74767	-73.97500	Entire home/apt	200	3	

To check the number of rows and columns present in the data.

```
In [5]: #to check the number of rows and column in dataset

        print("Air BNB : ")
        print("Rows : ", df.shape[0])
        print("Columns :", df.shape[1])

        Air BNB :
        Rows :  48895
        Columns : 16
```

In our data-set there are 48895 rows and 16 columns present.

Let's look into the details like datatypes for each columns using info().

```
In [6]: # to check the dataset summary

        df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   id                              48895 non-null  int64
 1   name                            48879 non-null  object
 2   host_id                         48895 non-null  int64
 3   host_name                       48874 non-null  object
 4   neighbourhood_group             48895 non-null  object
 5   neighbourhood                   48895 non-null  object
 6   latitude                        48895 non-null  float64
 7   longitude                       48895 non-null  float64
 8   room_type                       48895 non-null  object
 9   price                           48895 non-null  int64
 10  minimum_nights                  48895 non-null  int64
 11  number_of_reviews               48895 non-null  int64
 12  last_review                     38843 non-null  object
 13  reviews_per_month               38843 non-null  float64
 14  calculated_host_listings_count  48895 non-null  int64
 15  availability_365                48895 non-null  int64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB
```

2. Data Cleaning:

- Missing values

Next checking the null value count and percentage of null values in those column

```
In [8]: #Checking the null value count and percentage of null values in those column
        df.isnull().sum()
```

```
Out[8]: 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
```

```
In [9]: #Calculating the null value percentage

        def nul_val(df):
            miss = pd.DataFrame(columns = ['variable','percentage'])
            for clm in df.columns:
                if df[clm].isna().values.any():
                    percent = ((df[clm].isna().sum()/df.shape[0])*100).round(2)
                    miss = miss.append({'variable':clm, 'percentage':percent}, ignore_index = True)
            return miss
```

```
In [10]: null_df = nul_val(df)
         print(null_df)

                  variable  percentage
         0            name        0.03
         1       host_name        0.04
         2     last_review       20.56
         3  reviews_per_month       20.56
```

Missing value treatment:

As observed in the "last_review" and "reviews_per_month" columns we have 20.56% missing values. Since our end objective is to perform data analysis & try to gain insight for business hence we will not impute with mean or median values. Instead, we will zero in on null values to make our analysis easier

```
In [11]: # Replacing the null values in 'reviews_per_month' column with 0 to make analysis easier

         df.fillna({'reviews_per_month':0}, inplace=True)

In [12]: # Checking again for null values in 'reviews per month' column

         df['reviews_per_month'].isnull().sum()

Out[12]: 0
```

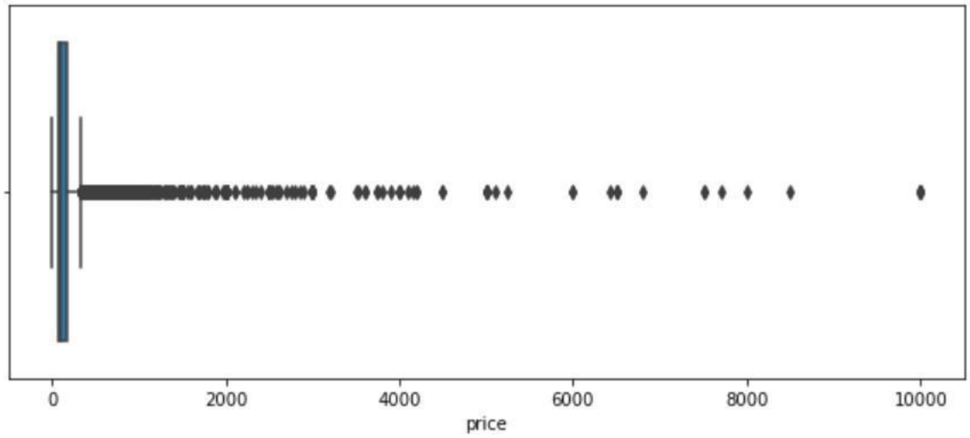
```
In [23]: # Replacing the null values in 'last_review' column with 0 to make analysis easier

         df.fillna({'last_review':0}, inplace=True)

In [24]: # Checking again for null values in 'last review' column

         df['last_review'].isnull().sum()

Out[24]: 0
```

Outlier analysis and treatment:

All numerical columns are subjected to Outlier Analysis to identify any outliers that need to be excluded.

Price column analysis:

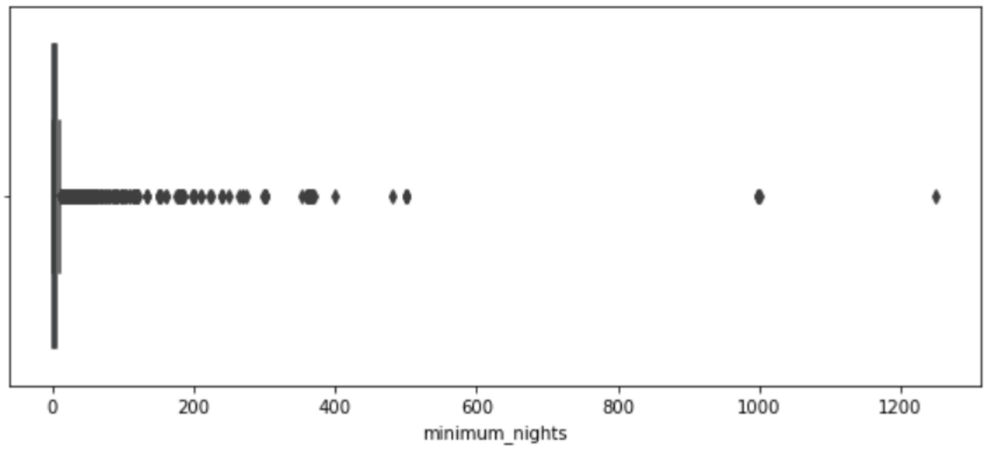
In [14]:
```
# Checking Outliers in price column

plt.figure(figsize = (10,4))
sns.boxplot(df['price'])
plt.show()
```



As we can see, there are outliers in the price column. In our analysis, this outlier can be useful for gaining insight into price distribution from a business perspective.

Minimum nights Analysis:

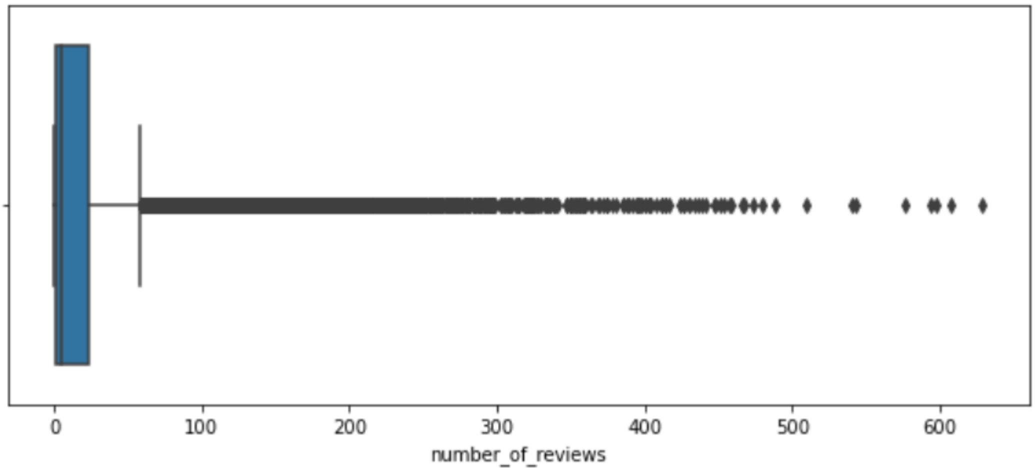
In [15]:
```
# Checking outliers in minimum nights column

plt.figure(figsize = (10,4))
sns.boxplot(df['minimum_nights'])
plt.show()
```



As observed, there are few outliers in Minimum Nights column as well. We will keep them intact since they can be useful from a business perspective.

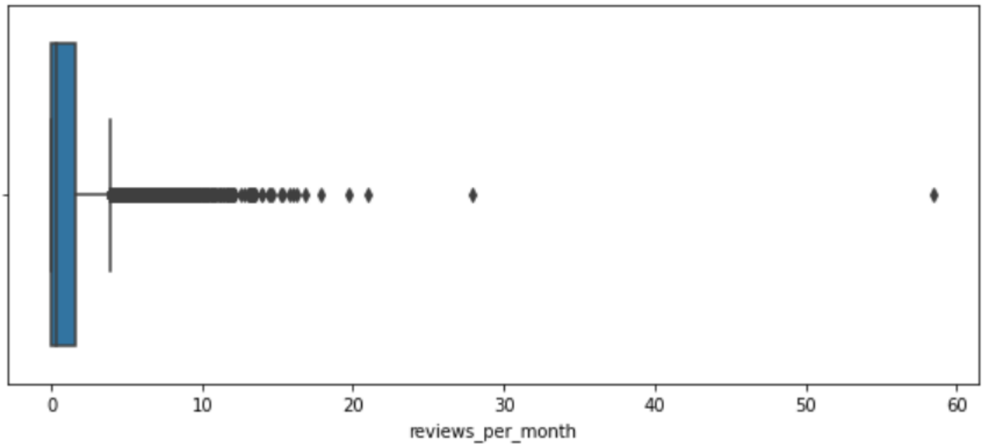
Reviews Analysis:

In [16]:
```
# Checking outliers in number of reviews column

plt.figure(figsize = (10,4))
sns.boxplot(df['number_of_reviews'])
plt.show()
```



Reviews per month Analysis:

In [17]:
```
# Checking outliers in reviews per month column

plt.figure(figsize = (10,4))
sns.boxplot(df['reviews_per_month'])
plt.show()
```

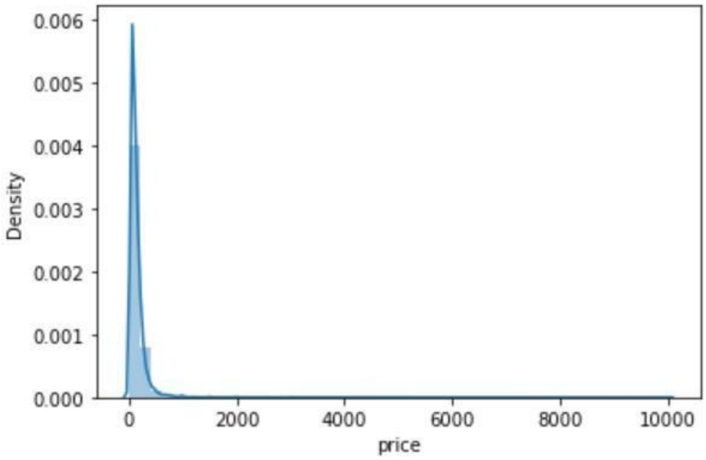


It was expected that there would be outliers in the number of reviews & reviews per month column, as some properties are quite popular among visitors and more people prefer to stay there. Hence, those properties have received more reviews than others. It will be helpful in our data analysis.

3. **Exploratory Data**

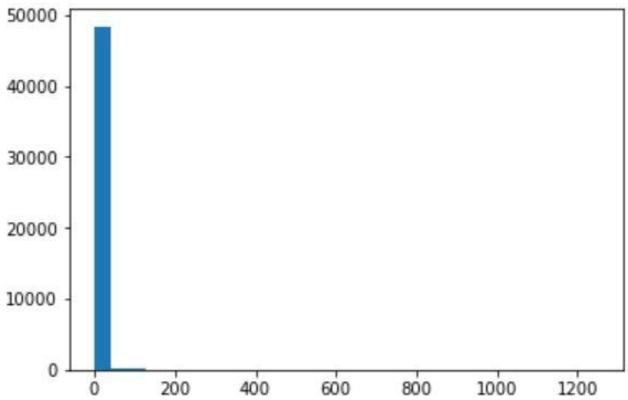
Univariate Analysis:

```
In [18]: # Plotting histogram for Price column

         sns.distplot(df['price'])

Out[18]: <AxesSubplot:xlabel='price', ylabel='Density'>
```



From the above plot, we can see most of the property price falls between 0 and 1000, with few properties ranging from 2000 to 10000 which are an outlier.

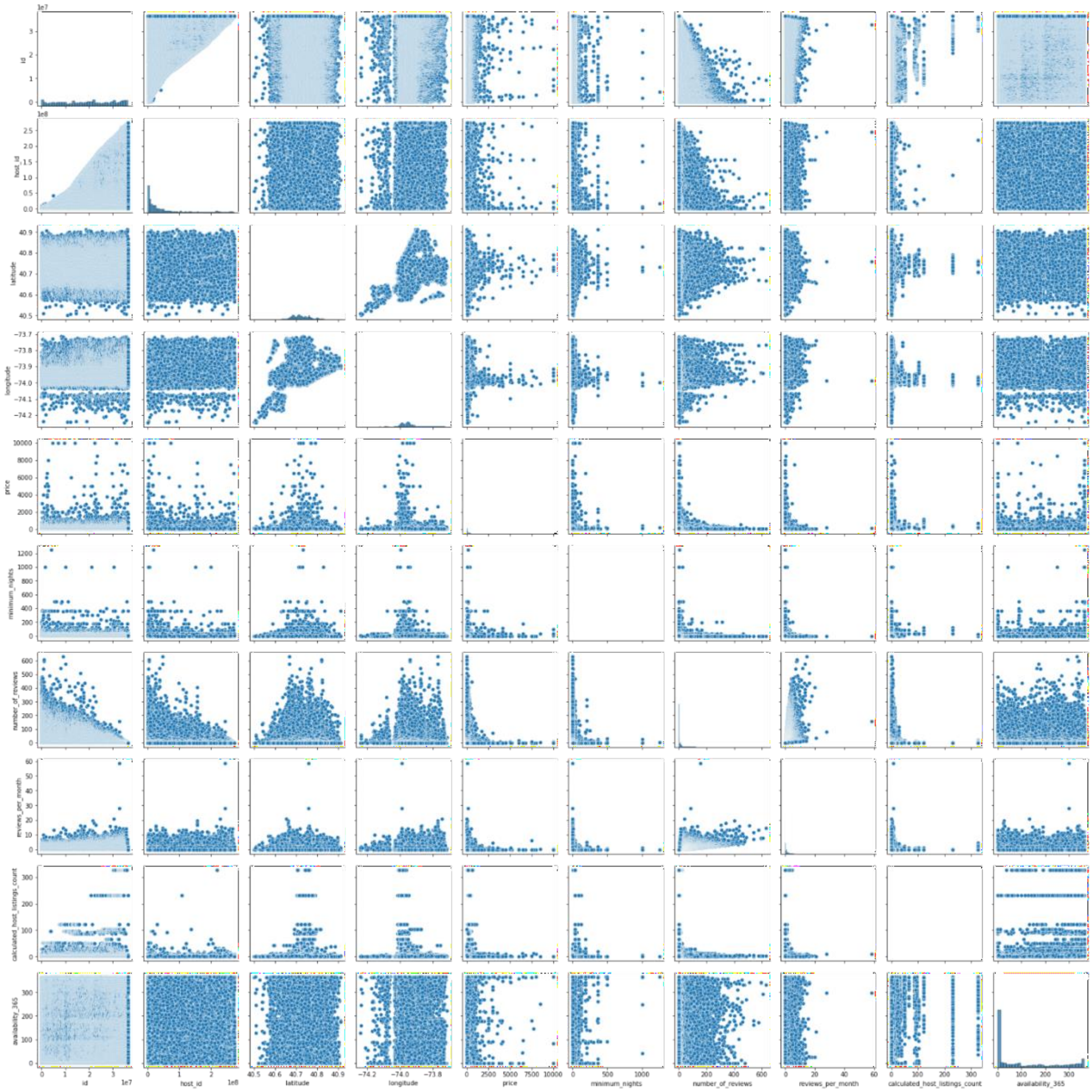
```
In [19]: # Minimum Night distribution

         plt.hist(df['minimum_nights'],bins = 30)
         plt.show()
```



Based on the above plot, most of the properties offer a minimum stay of 0 to 6 days.

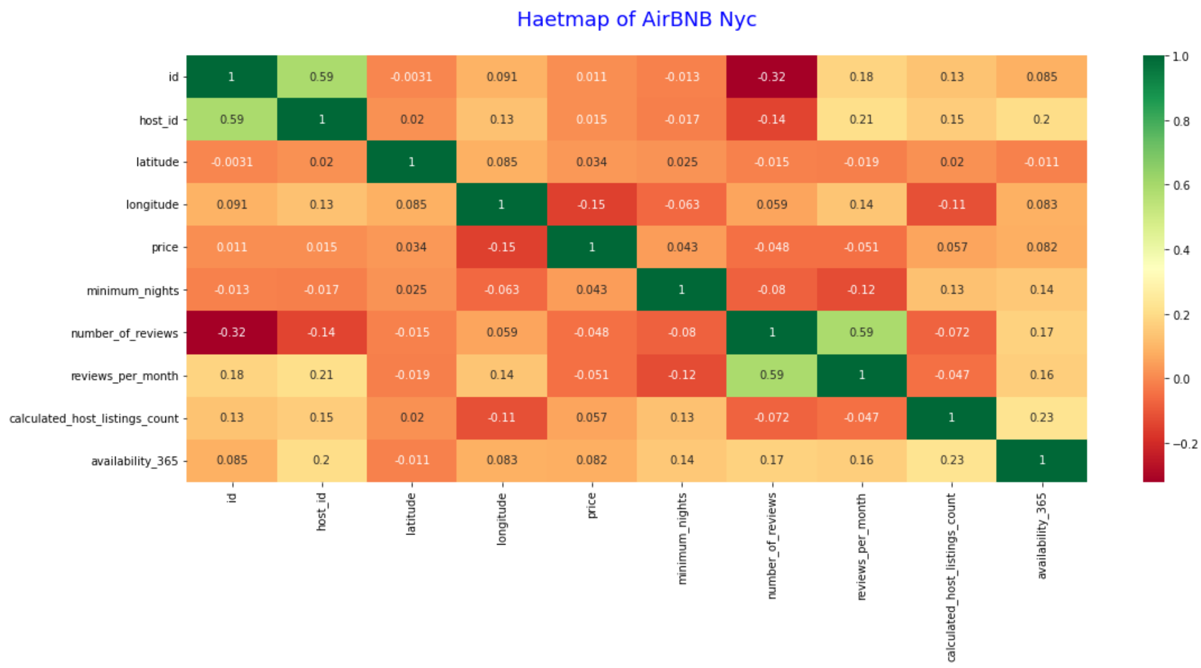
<u>Bivariate Analysis:</u>

We use pair plot to understand the correlation between the numerical columns.



Heatmap of AirBnB data:

In [21]:
```
# Heat-map for coorelation

plt.figure(figsize = [18,7])
sns.heatmap((df.corr()), annot = True, cmap = 'RdYlGn')
plt.title("Haetmap of AirBNB Nyc\n", color = "Blue", fontsize = 18)
plt.show()
```



From our analysis we have observed that there is a negative correlation between price, minimum nights and number of reviews. And on the other hand, we can see there is a positive correlation between Calculated_host_listings_count and minimum_nights & availability _365 columns.

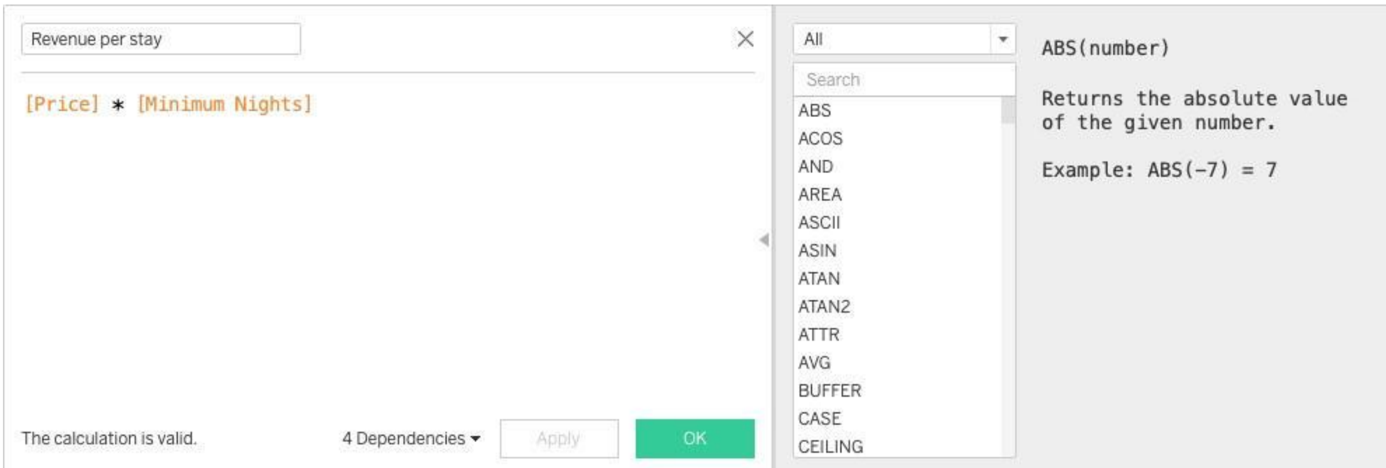
Data Visualization and Analysis using Tableau:

We have used tableau to visualize the data for the assignment.

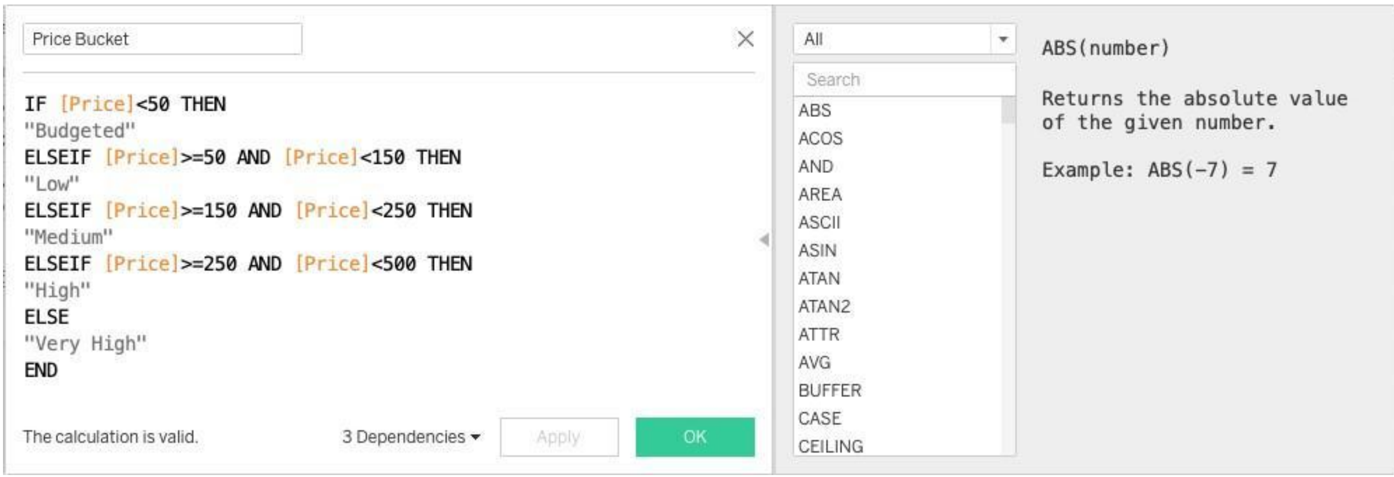
We will use Tableau for Data Visualization and Analysis to come up with Insights and observations. Recommendations are made from the insights and observations drawn from the Analysis.

Derived Column Calculation

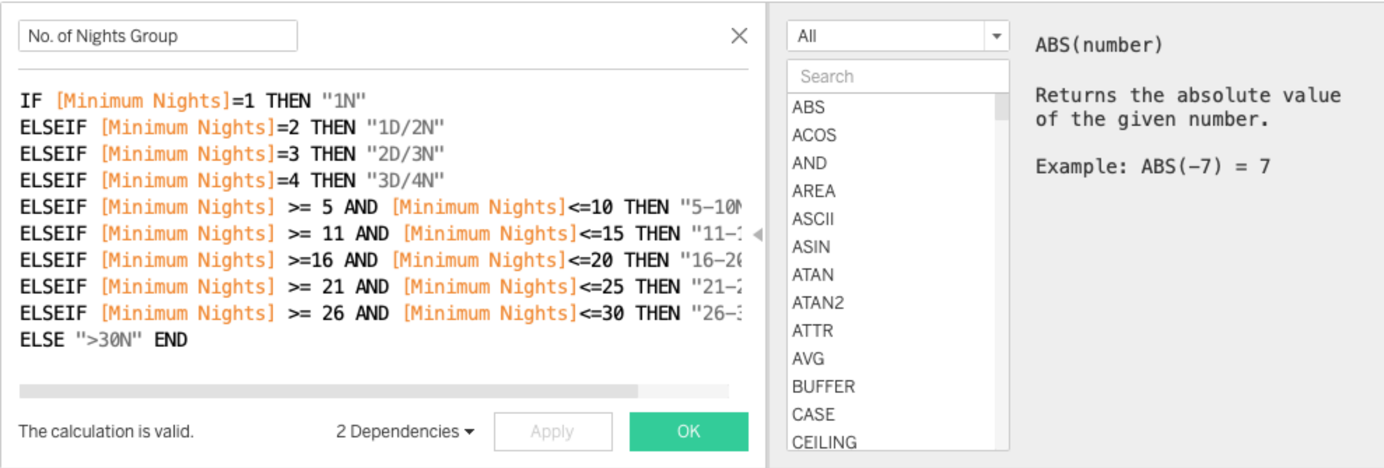
Revenue Per Stay:



Price Bucket:

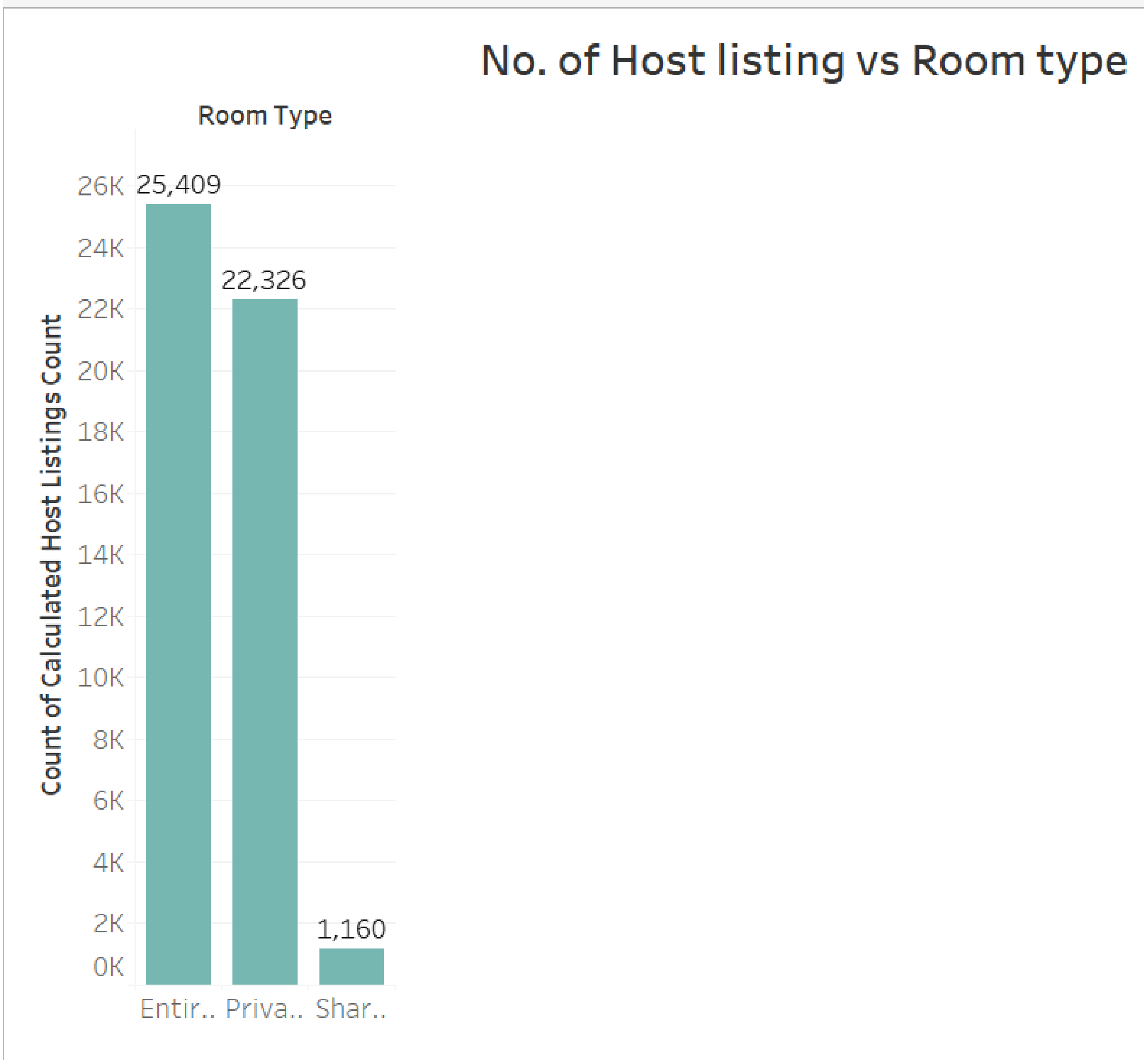


Number of Nights Group:



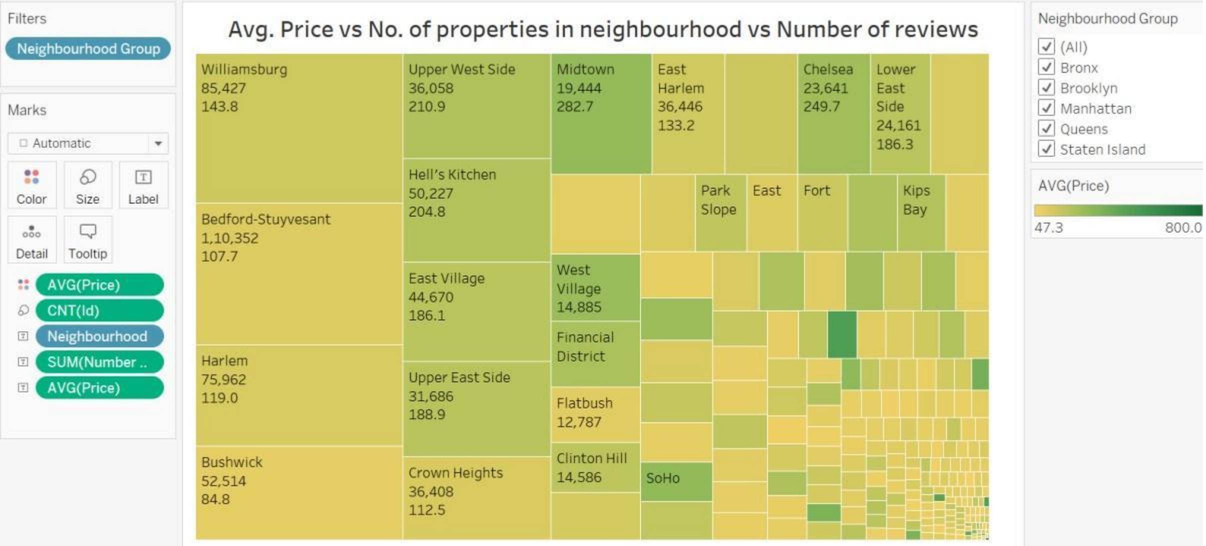
Analysis & Insights:

Properties based on Neighbourhood Group and Room Type.



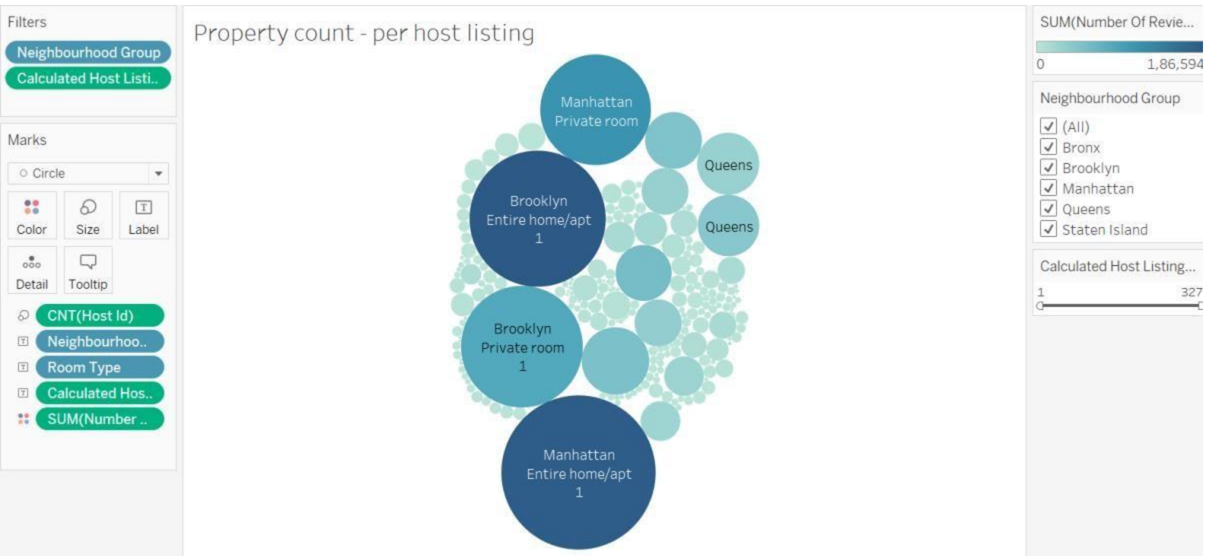
Observation: About 96% of the properties falls under Entire Home/apt. category. Private rooms are the second largest category and very few properties are listed under Shared room across all the Neighbourhood.

Neighbourhood with Median Price and Number of Reviews



Observation: The general trend of booking is majorly focused on Price. Here, a number of reviews (booking) are more for lower-priced properties. We can see a contrary trend also for some properties with high prices got more bookings too. So some people are okay with the price if good facilities may be offered.

Property count in neighbourhood



Observation: As we can see Brooklyn & Manhattan neighbour-hood group has the highest number property in Entire Home/Apt category followed by Private room.

Popular Price Range in Neighbourhood Groups Vs Room Type

Observations: The popular price ranges are from 50 to 250 range. Above 250 , not many booking are seen.

Insight: Most listed properties falls under the medium and low price category. And maximum properties are located in Manhattan & Brooklyn neighbourhood.

Price spread for different Room Type

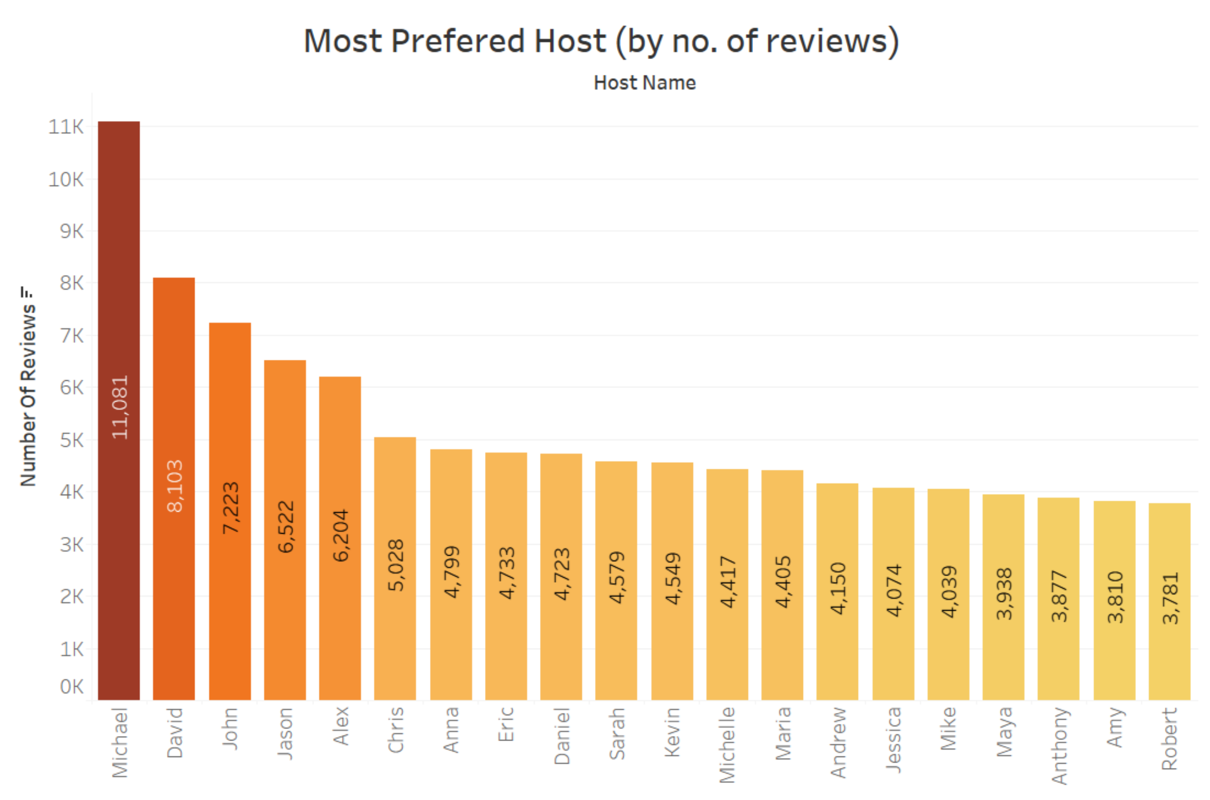
Price Range vs Room Type

Price Bucket

Room Type	Budget..	High	Low	Medium	Very Hi..
Entire home/apt	117	4,569	10,220	9,453	1,050
Private room	4,291	399	16,201	1,261	174
Shared room	619	23	462	45	11

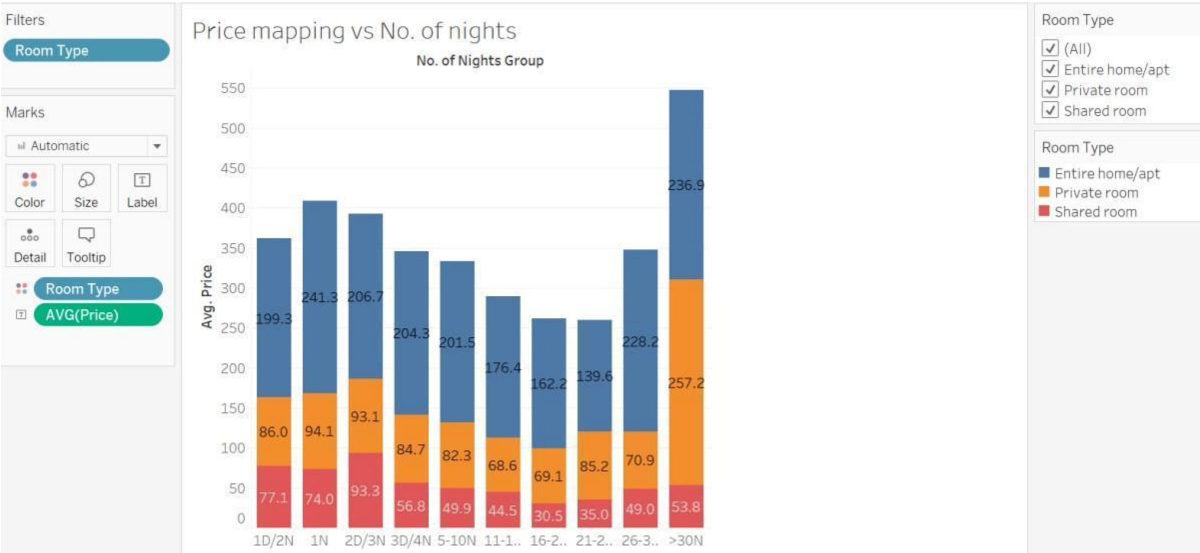
Observation: Entire Home/Apt is costly compared to Private rooms and shared rooms. People prefer mostly Entire Home or Private rooms based on the availability.

Most preferred Host(by no: of reviews)



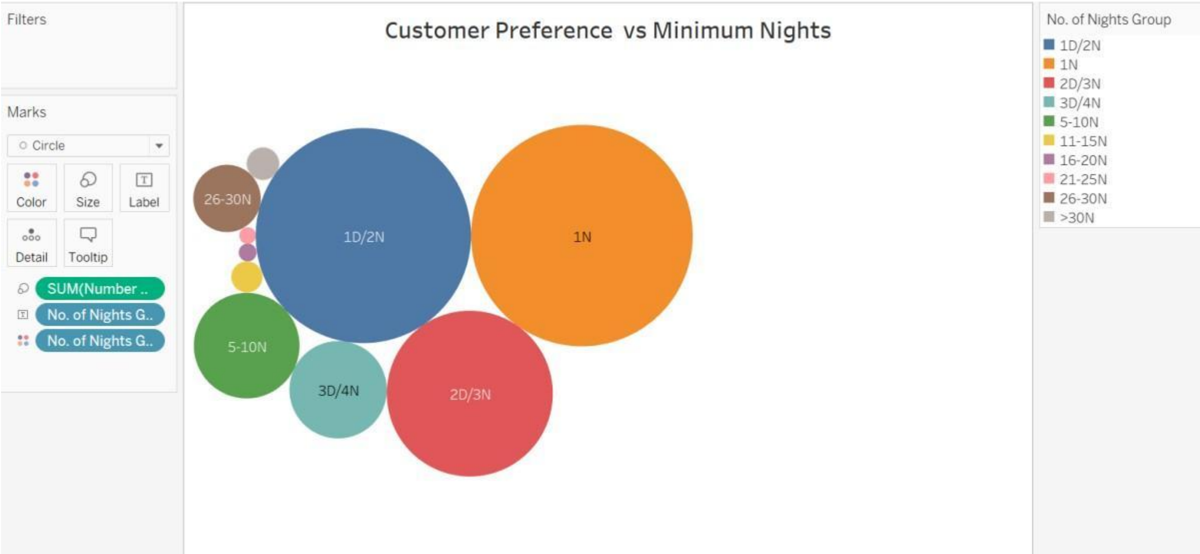
Observation/Insights: Based upon the number of reviews received we have identified top 20 most preferred host. Micheael has received the maximum number of reviews indicating that he provided good stay experience to the visitors

Minimum Nights vs Price



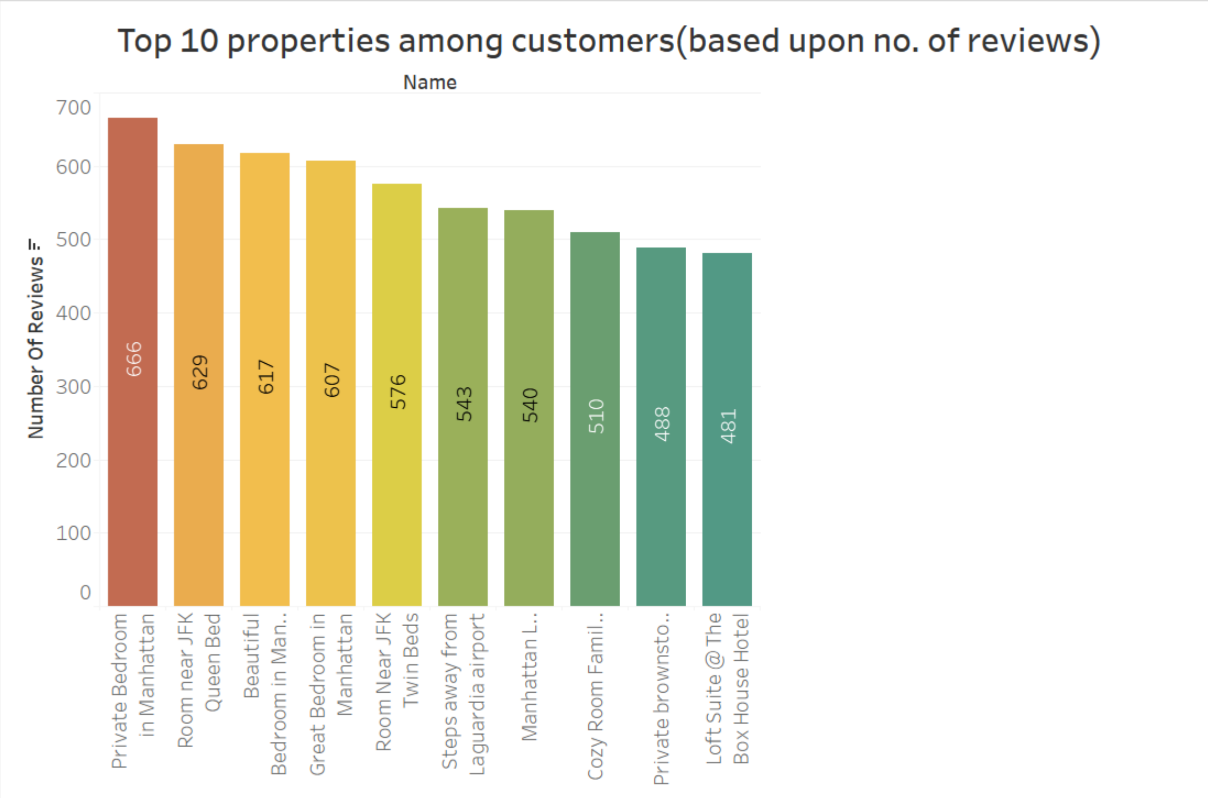
Observation: The average price for Entire Home/Apt show the similar trend across the minimum nights required for booking. However, Private room category has almost equivalent average price for long duration stay (>30 days)The hike in the revenue is for Minimum stays 1 to 5 days , mainly 1, 2 and 3 night stay and 30 days. Business can think about the properties having min night stays in these categories across all locations.

Minimum Nights Distribution



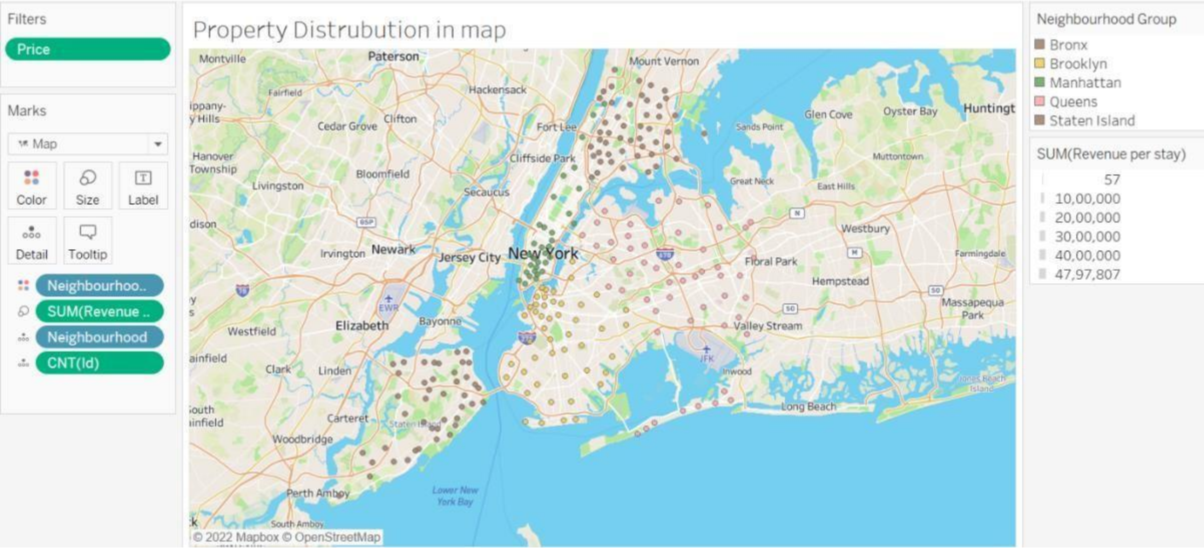
Observation: Maximum Properties are for Minimum night stay 1,2 and 3 days and quite few property offer minimum nights required for booking more than 4 nights for all the neighbourhood groups and Room Type.

Top 10 Properties (based upon number of reviews)



Observation : We have identified the Top 10 properties based upon the number of reviews received by them. And found Private Room in Manhattan is the most preferred property for stay among the visitors.

Property distribution in Map



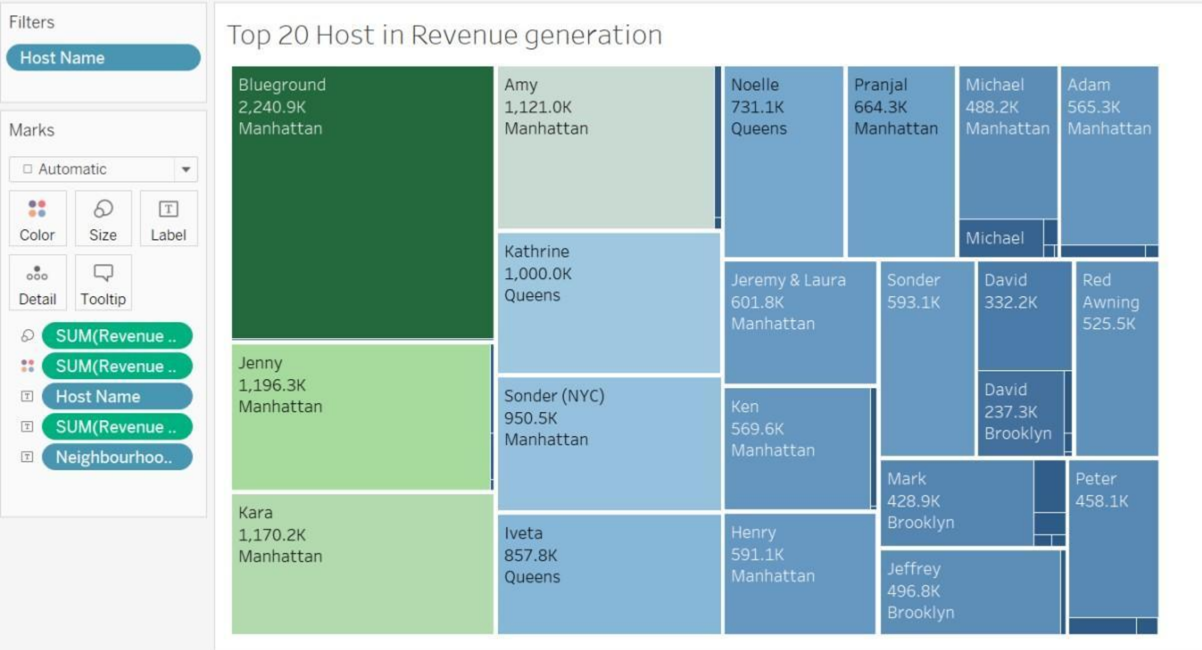
Observation: It is clearly seen that the median price for properties are affordable except few properties. Manhattan and Brooklyn at an average, price is medium to high ranges compared to Bronx and Queens. In Staten Island we see the median price is higher in some areas like Woodrow and FortWadsworth than other regions like Manhattan and Brooklyn.

Property Availability distribution with revenue

Observation: The above figure represents the highest revenue is generated for the property where Availability is "0" compared to other Availability_365 options.

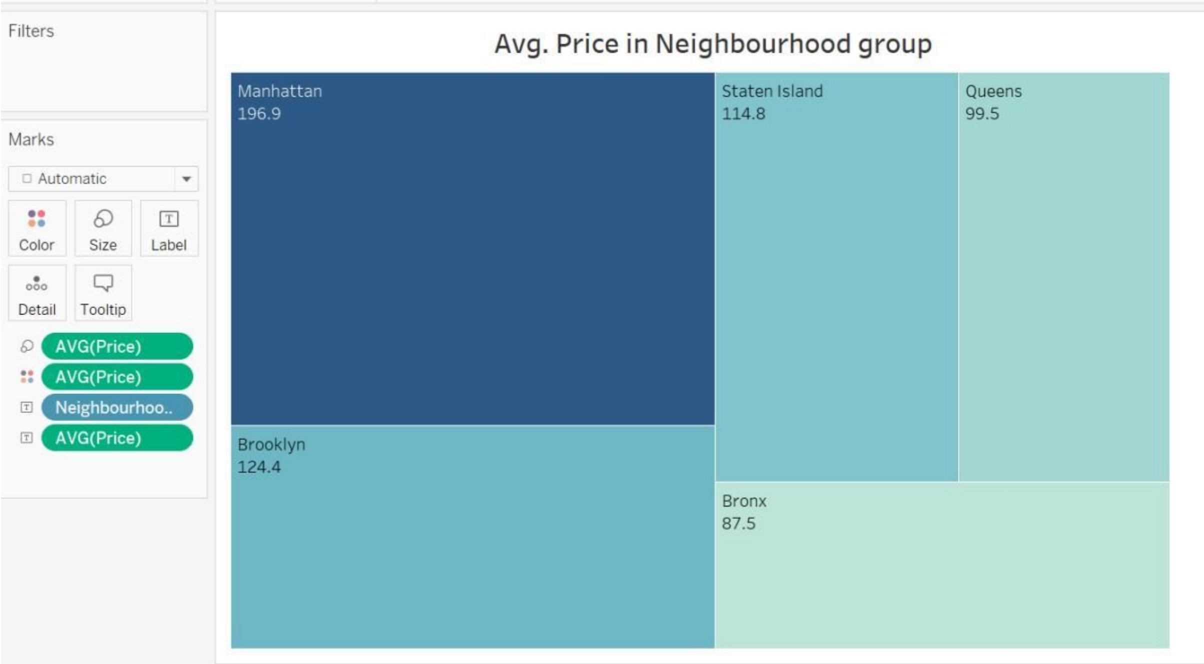
Insight: It is possible that many visitor or people are interested in staying for less number of nights. So by reducing the minimum nights required for booking for most of the property in order to increase the revenue. Assuming that, we can restart the service with less minimum nights required in order to increase the visitor and revenue.

Top 20 Earning Hosts in revenue generation:



Observations: Above chart shows the top 20 host who earn highest revenues. And host’s from Manhattan has the highest contribution towards adding the revenue.

Neighbourhood group Vs Avg. Price



Observations: We can observe that the avg rate of properties in Manhattan across all category is higher compared to other neighbourhoods.