Storytelling Case study: Airbnb, NYC

Methodology Document

By – Shubham Saurav

Kamna Sethi

Ajit Shetty

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.

The different leaders at Airbnb want to understand some important insights based on various attributes in the data set so as to increase the revenue.

Analysing the Dataset:

1. Firstly analysis of dataset is done on jupyter notebook.

```
#Import the required Libraries.

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

#import the warnings.

import warnings
warnings.filterwarnings("ignore")

pd.set_option("display.max_columns",None)
pd.set_option("display.max_rows",None)
```

```
df = pd.read_csv('AB_NYC_2019.csv')
df.head()
```

	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	

```
df.shape
```

```
(48895, 16)
```

```
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
```

- After loading dataset we can see that there are total of 48895 rows and 16 columns.
- There are various attributes related to Airbnb in the dataset such as name, host_id, neighbourhood, latitude, longitude, room_type, price etc.

2. There are null values in 4 columns. These columns are name, host_name, last_review and reviews_per_month.

```
df.isnull().mean()*100
```

```
id                                0.000000
name                              0.032723
host_id                           0.000000
host_name                         0.042949
neighbourhood_group               0.000000
neighbourhood                     0.000000
latitude                          0.000000
longitude                         0.000000
room_type                         0.000000
price                             0.000000
minimum_nights                    0.000000
number_of_reviews                 0.000000
last_review                      20.558339
reviews_per_month                20.558339
calculated_host_listings_count    0.000000
availability_365                  0.000000
dtype: float64
```

3. Handling missing values: Missing values are replaced by either median or mode values.

```
rpm_med = df['reviews_per_month'].median()
df['reviews_per_month'] = df['reviews_per_month'].fillna(rpm_med)
```

```
df['last_review'].fillna(df['last_review'].mode()[0],inplace = True)
```

```
df['name'].fillna(df['name'].mode()[0],inplace = True)
```

```
df['host_name'].fillna(df['host_name'].mode()[0],inplace = True)
```

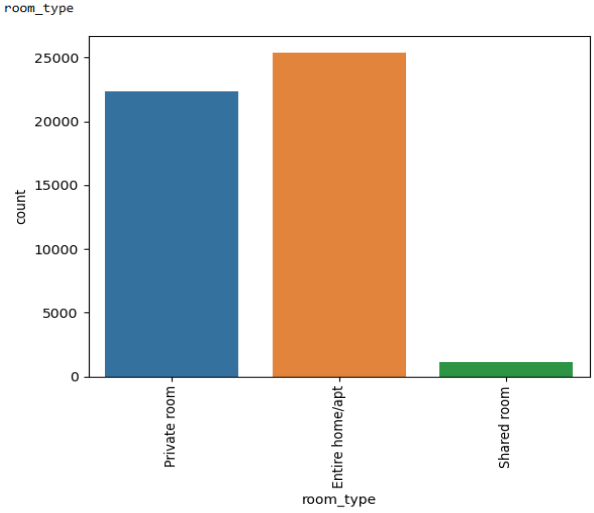
```
df.isnull().mean()*100
```

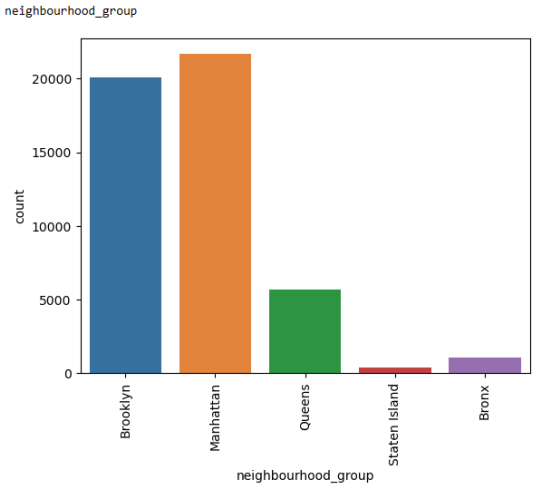
```
id                                0.0
name                              0.0
host_id                           0.0
host_name                         0.0
neighbourhood_group               0.0
neighbourhood                     0.0
latitude                          0.0
longitude                         0.0
room_type                         0.0
price                             0.0
minimum_nights                    0.0
number_of_reviews                 0.0
last_review                       0.0
reviews_per_month                 0.0
calculated_host_listings_count    0.0
availability_365                  0.0
dtype: float64
```

4. Analysing different columns :

- In room_type category Entire home\apt has the highest count.
- In neighbourhood_group Manhattan has the highest count followed by Brooklyn.

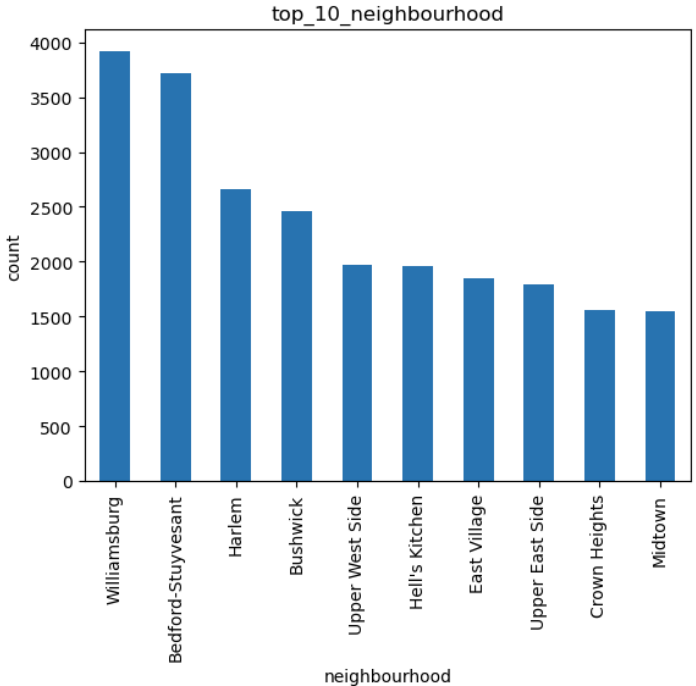
```
In [36]: for i in cat_var:
             print(i)
             sns.countplot(x=df[i])
             plt.xticks(rotation = 90)
             plt.show()
```





- Here is the list of top 10 neighbourhood where Williamsburg has the highest count, followed by Bedford-Stuyvesant.

```
In [50]: top_10_neighbourhood = df.neighbourhood.value_counts().head(10)
         top_10_neighbourhood.plot.bar()
         plt.title('top_10_neighbourhood')
         plt.xlabel('neighbourhood')
         plt.ylabel('count')
         plt.show()
```



```
df['minimum_nights'].quantile([0,0.25,0.50,0.75,0.90,0.95,0.99,0.997])
```

```
0.000      1.0
0.250      1.0
0.500      3.0
0.750      5.0
0.900     28.0
0.950     30.0
0.990     45.0
0.997    120.0
Name: minimum_nights, dtype: float64
```

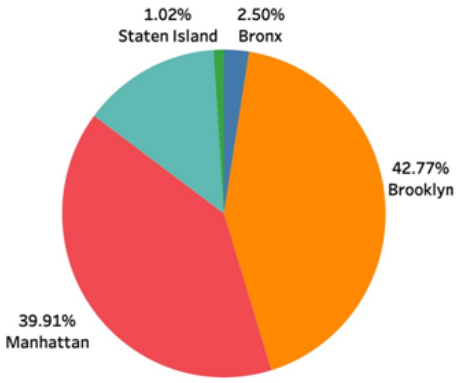
- Binning the minimum nights for further analysis in tableau.

```
df.to_csv("AB_NYC_updated.csv", index=False)
```

```
DataFrame is written to Excel File successfully.
```

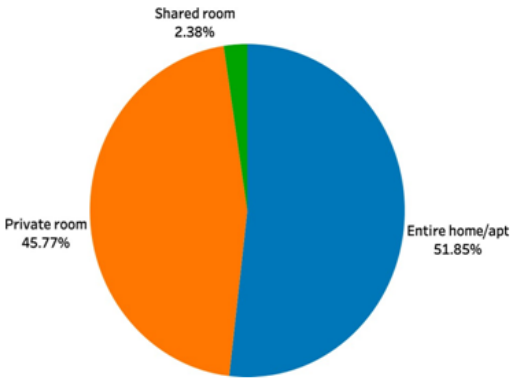
After cleaning and data modification we wrote the data to another CSV and used it in tableau for further analysis.

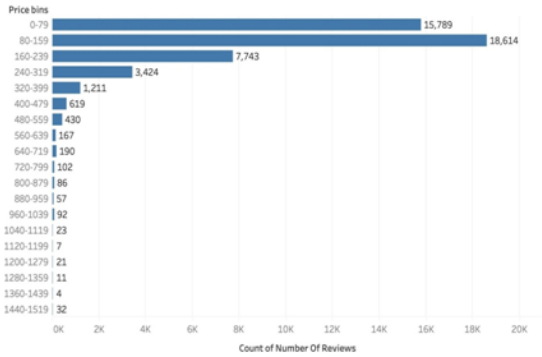
Tableau analysis



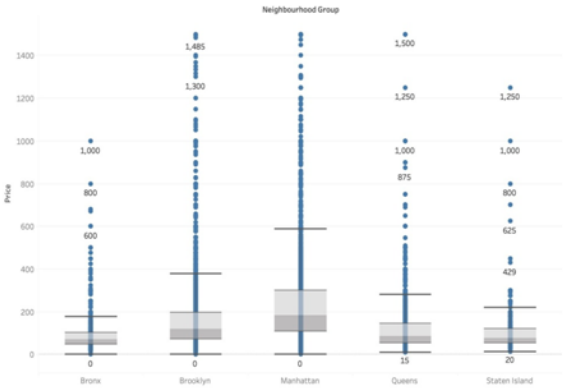
Top 2 most preferred neighbourhood group Brooklyn and Manhattan with Brooklyn having 42.77 total reviews

The highest number of property booked is Entire home/apt followed by Private room



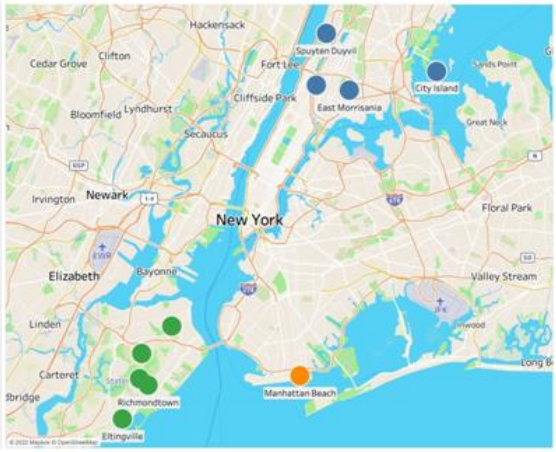


Customers prefer stays that comes in range of 80-159 USD across NYC, followed by 0-79 USD.

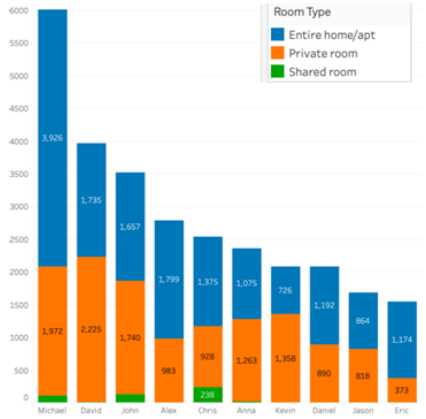


Most and least expensive borough is Manhattan and Bronx with 184 and 70 USD median.

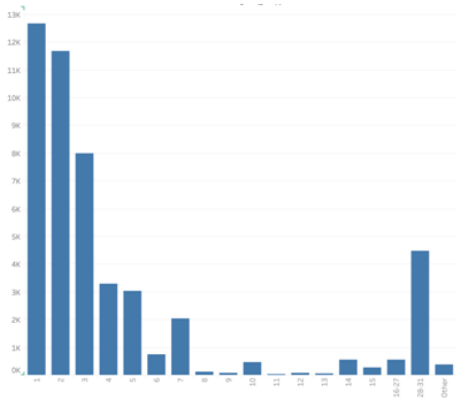
Entire room/ Apt. are the highest revenue generator.



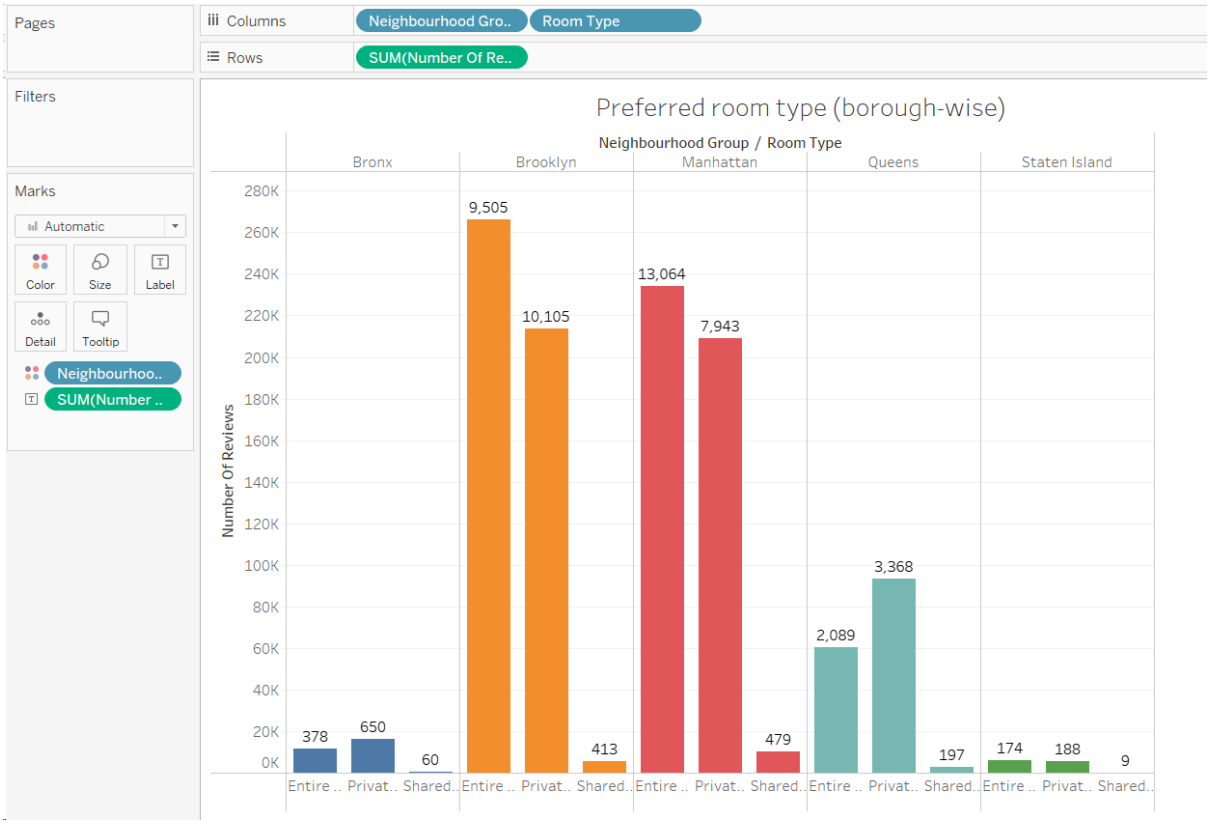
The neighbourhood that generate highest revenue are near to some water body



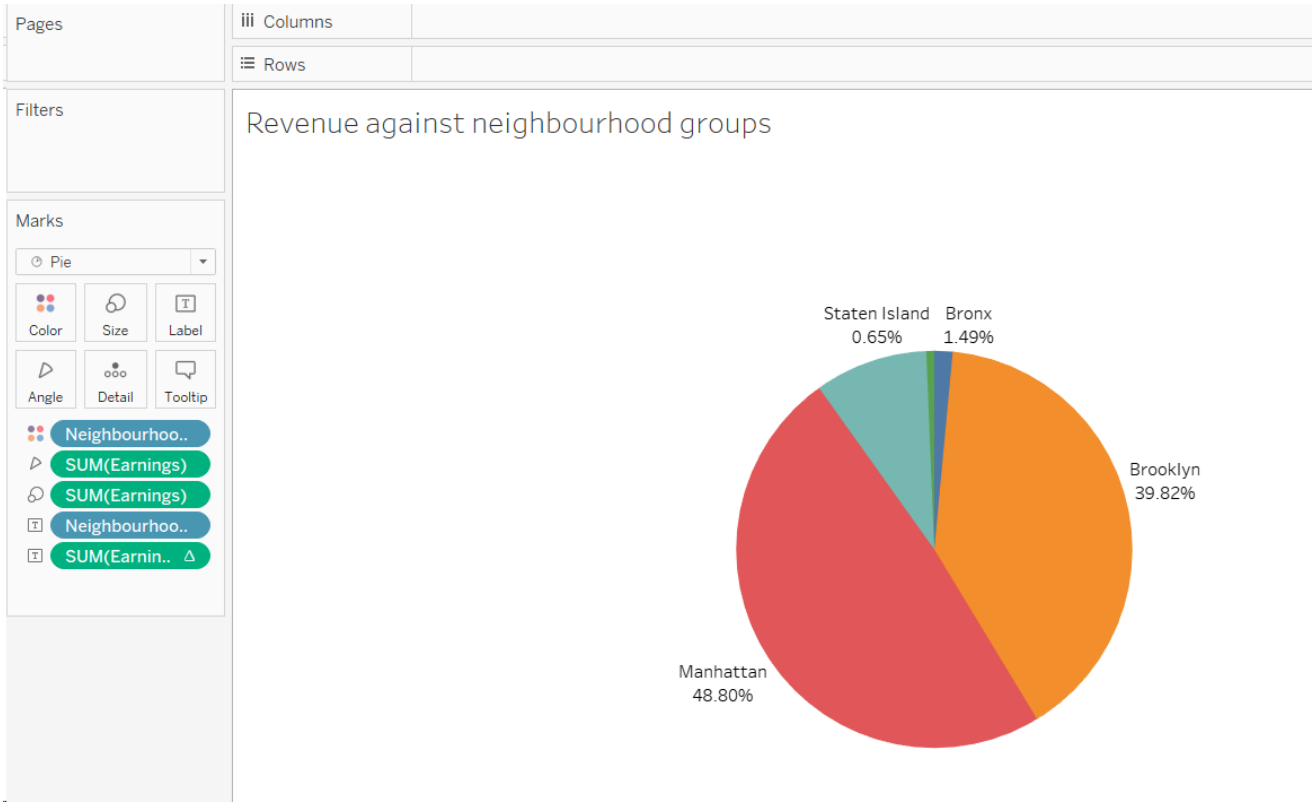
Most popular overall host is Michael which is in Manhattan



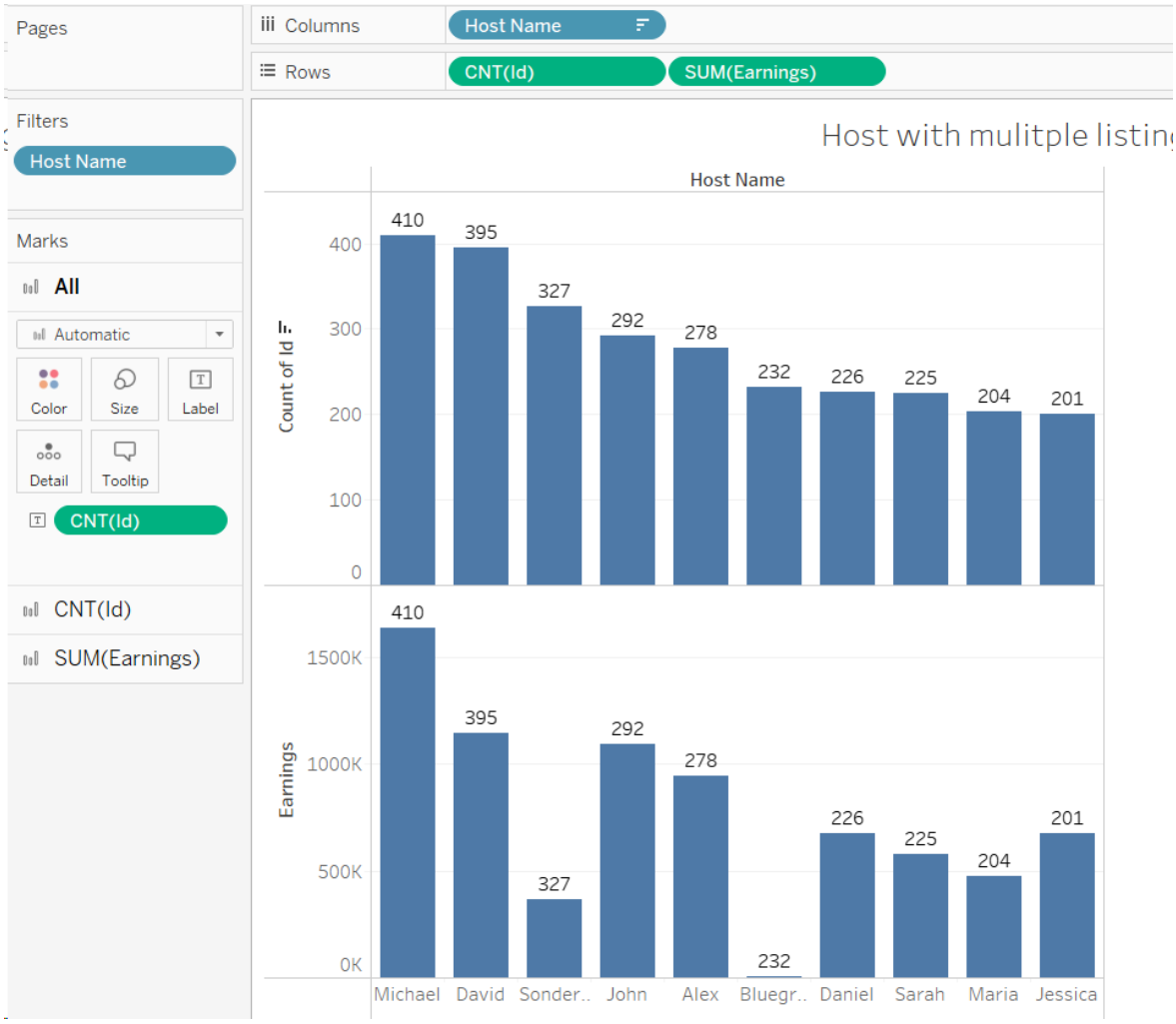
The highest number of reviews is given by 1 night stays and a spike was seen in reviews for stay of 28-31 days

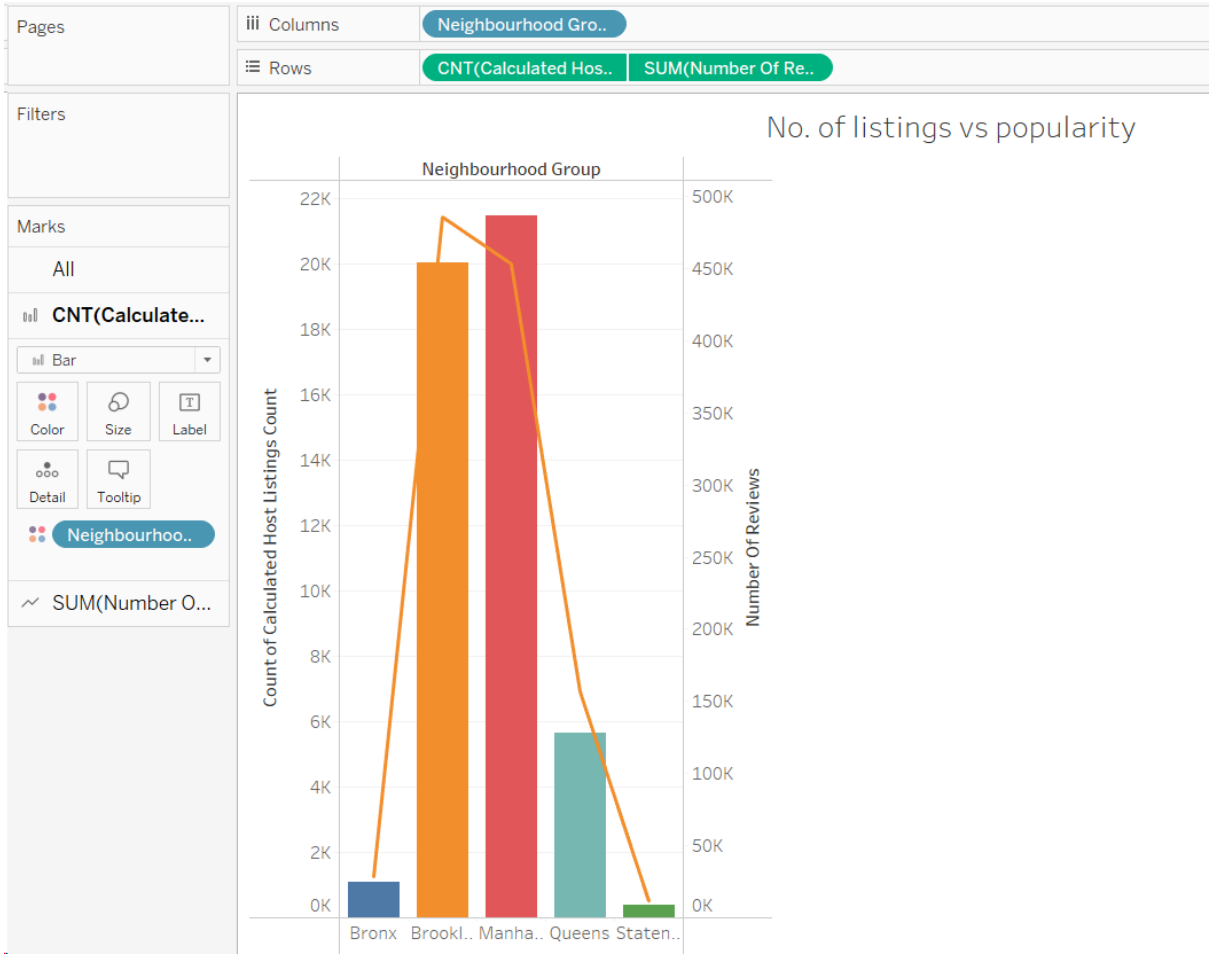


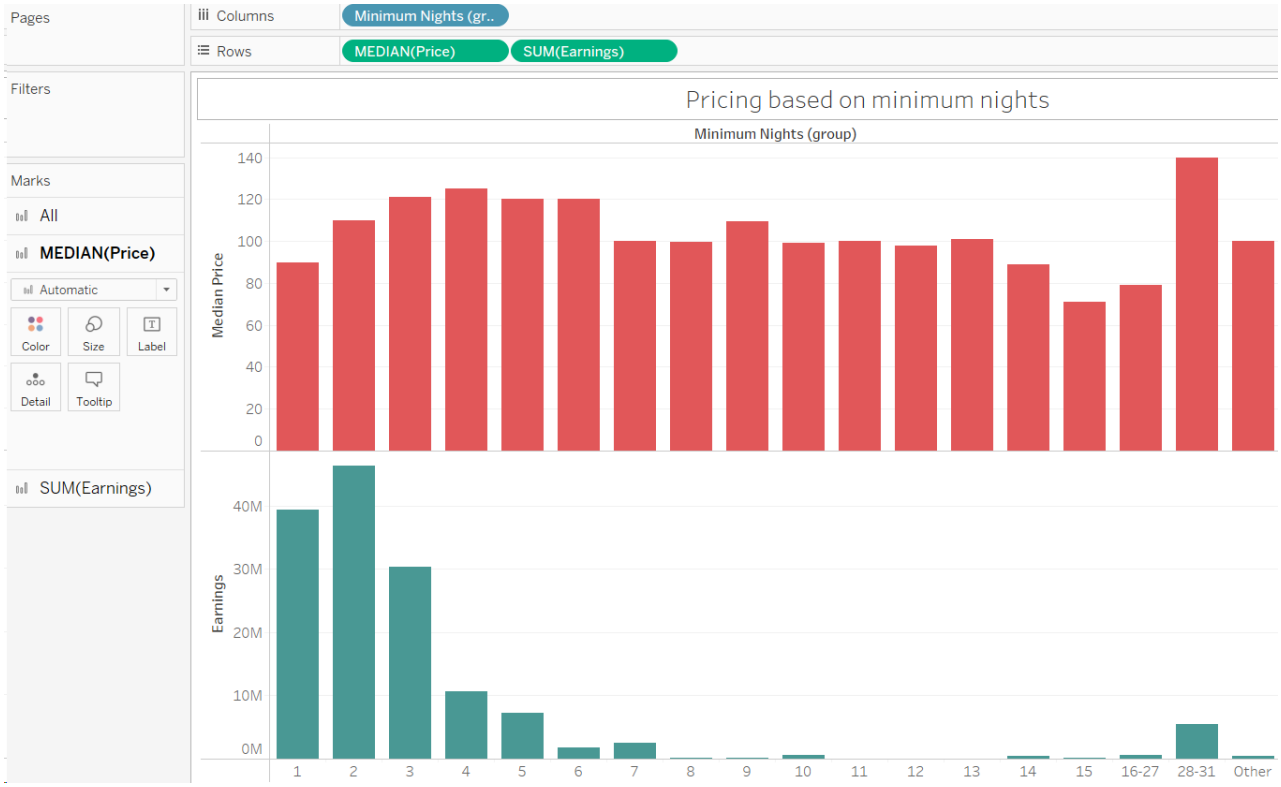
- Increasing the number of Entire home/apartment and private room in Queens and Bronx will helps to attract more customers as it is closer to Manhattan.

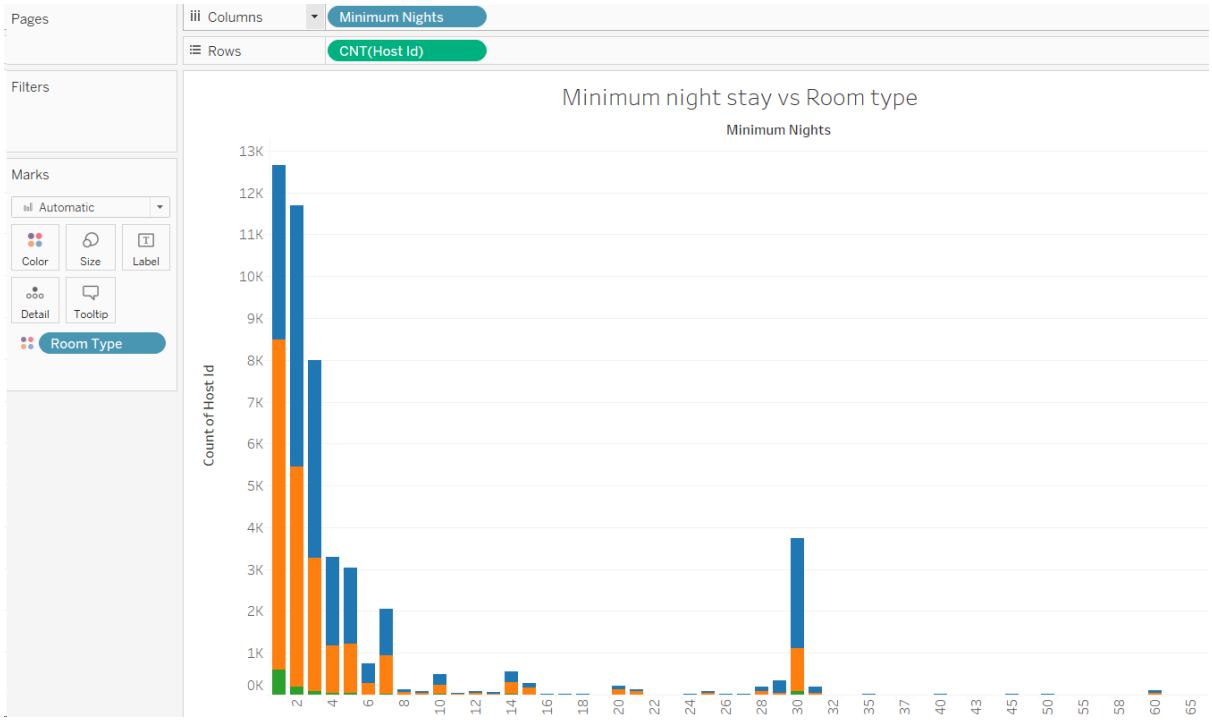


- ❑ Listings should be increased in Staten Island especially the sea-side view
- ❑ In Manhattan and Brooklyn shared room type should be increased as it has lower price, scenic view and would attract more customers.

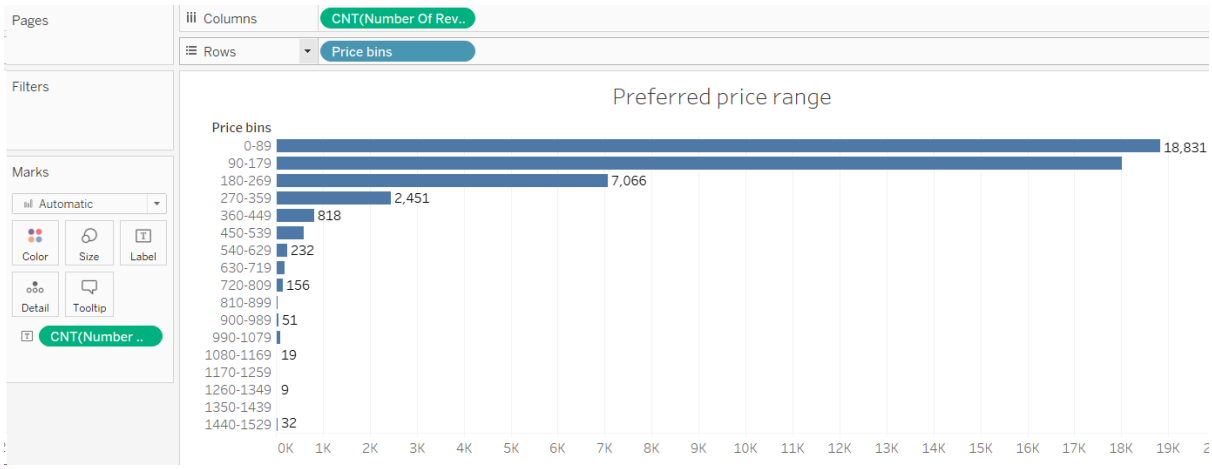

Pages
Columns
Host Name
Rows
CNT(Id)
SUM(Earnings)
Filters
Host Name
Marks
All
Automatic
Color
Size
Label
Detail
Tooltip
CNT(Id)
CNT(Id)
SUM(Earnings)
Host with mulitple listing
Host Name
Count of Id
410
395
327
292
278
232
226
225
204
201
Earnings
410
395
292
278
327
232
226
225
204
201
Michael
David
Sonder..
John
Alex
Bluegr..
Daniel
Sarah
Maria
Jessica


Pages
Columns
Neighbourhood Gro..
Rows
CNT(Calculated Hos..
SUM(Number Of Re..
Filters
Marks
All
CNT(Calculate...
Bar
Color
Size
Label
Detail
Tooltip
Neighbourhoo..
SUM(Number O...
No. of listings vs popularity
Neighbourhood Group
Count of Calculated Host Listings Count
Number Of Reviews
Bronx
Brookl..
Manha..
Queens
Staten..


Pages
Columns
Minimum Nights (gr..
Rows
MEDIAN(Price)
SUM(Earnings)
Filters
Marks
All
MEDIAN(Price)
Automatic
Color
Size
Label
Detail
Tooltip
SUM(Earnings)
Pricing based on minimum nights
Minimum Nights (group)
Median Price
140
120
100
80
60
40
20
0
Earnings
40M
30M
20M
10M
0M
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16-27
28-31
Other


Pages
Columns
Minimum Nights
Rows
CNT(Host Id)
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Room Type
Minimum night stay vs Room type
Minimum Nights
Count of Host Id
13K
12K
11K
10K
9K
8K
7K
6K
5K
4K
3K
2K
1K
0K
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
35
37
40
43
45
50
55
58
60
65

- ❑ The highest number of property booked is Entire home/apt followed by Private room.
- ❑ The median price is almost the same for night stays under 7days which might discourage customers to book Entire home/apt or private room type in Manhattan and Brooklyn



- ❑ Higher number of properties does not promise higher earning.
- ❑ Bronx and Staten Island are gaining popularity very rapidly.
- ❑ With the above findings we can declutter the host earning.

Recommendation

- ❑ Prime Location: In Manhattan and Brooklyn shared room type should be increased as it has lower price, scenic view and would attract more customers.
- ❑ Market Traction: Increasing the number of Entire home/apartment and private room in Queens and Bronx will helps to attract more customers as it is closer to Manhattan.

- ❑ Listings should be increased in Staten Island especially the sea-side view
- ❑ Pricing: Preferred number of night stay is up to a 7 nights and 28-31 nights. Therefore, introducing offers for such customers can help in increasing the revenue.
- ❑ Encourage Host : Hosts should work on improving the quality of Entire home / apt and Private rooms to improve the property's review.