Assumptions and considerations:

In the case study we have used Jupiter notebook to perform initial analysis of the data and Tableau for data analysis and visualization.

Initial Analysis using Jupiter Notebook:

Data Set Used: AB_NYC.2019.csv
Number of Rows: 48895
Number of Columns: 16

Loading the Data ¶

In [2]:
```
df = pd.read_csv('AB_NYC_2019.csv')
df.head()
```

Out[2]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revi
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	

In [3]:
```
df.shape
```

Out[3]:
```
(48895, 16)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
id                                48895 non-null int64
name                              48879 non-null object
host_id                           48895 non-null int64
host_name                         48874 non-null object
neighbourhood_group               48895 non-null object
neighbourhood                     48895 non-null object
latitude                          48895 non-null float64
longitude                         48895 non-null float64
room_type                         48895 non-null object
price                             48895 non-null int64
minimum_nights                    48895 non-null int64
number_of_reviews                 48895 non-null int64
last_review                       38843 non-null object
reviews_per_month                 38843 non-null float64
calculated_host_listings_count    48895 non-null int64
availability_365                  48895 non-null int64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB
```

We have 48895 rows and 16 columns among which 3 are float type, 7 are int type and remaining 6 are object type.

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
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
```

So, we can see there are around 10k null values in the last_review and review_per_month columns and a very few null values in name and host_name. We are not handling the null values here.

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

Out[6]:

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listing
count	4.889500e+04	4.889500e+04	48895.000000	48895.000000	48895.000000	48895.000000	48895.000000	38843.000000	48895
mean	1.901714e+07	6.762001e+07	40.728949	-73.952170	152.720687	7.029962	23.274466	1.373221	7
std	1.098311e+07	7.861097e+07	0.054530	0.046157	240.154170	20.510550	44.550582	1.680442	32
min	2.539000e+03	2.438000e+03	40.499790	-74.244420	0.000000	1.000000	0.000000	0.010000	1
25%	9.471945e+06	7.822033e+06	40.690100	-73.983070	69.000000	1.000000	1.000000	0.190000	1
50%	1.967728e+07	3.079382e+07	40.723070	-73.955680	106.000000	3.000000	5.000000	0.720000	1
75%	2.915218e+07	1.074344e+08	40.763115	-73.936275	175.000000	5.000000	24.000000	2.020000	2
max	3.648724e+07	2.743213e+08	40.913060	-73.712990	10000.000000	1250.000000	629.000000	58.500000	327

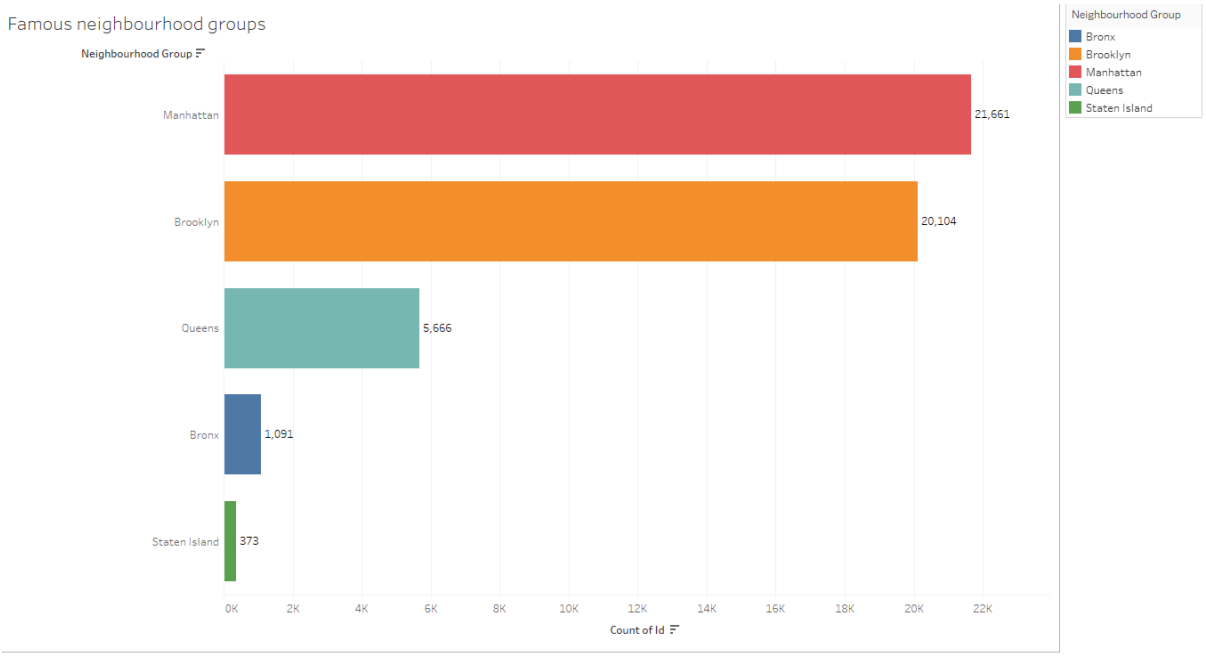
There are outliers found in few numeric columns but we are not removing any outliers as we are going to use medians in the analysis further.

In [7]:
```
df.nunique()
```

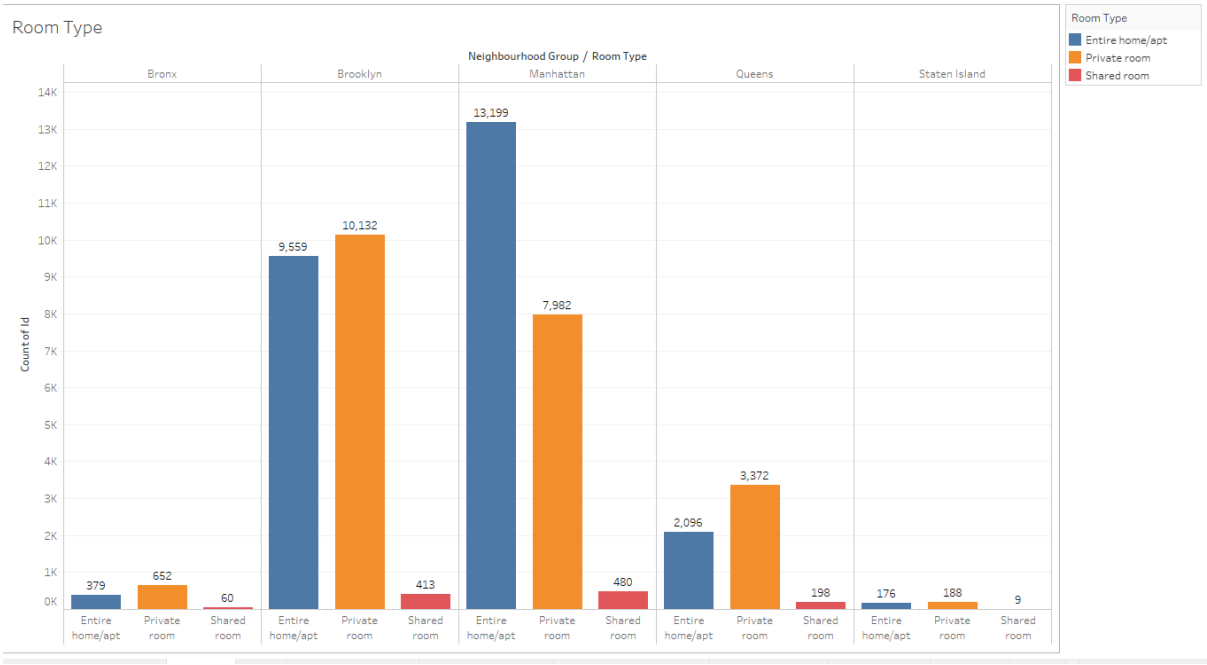
Out[7]:
```
id                                48895
name                              47896
host_id                           37457
host_name                         11452
neighbourhood_group                   5
neighbourhood                       221
latitude                          19048
longitude                         14718
room_type                             3
price                               674
minimum_nights                      109
number_of_reviews                   394
last_review                        1764
reviews_per_month                   937
calculated_host_listings_count       47
availability_365                    366
dtype: int64
```

Notable that we have 5 locations in dataset and 3 room types.

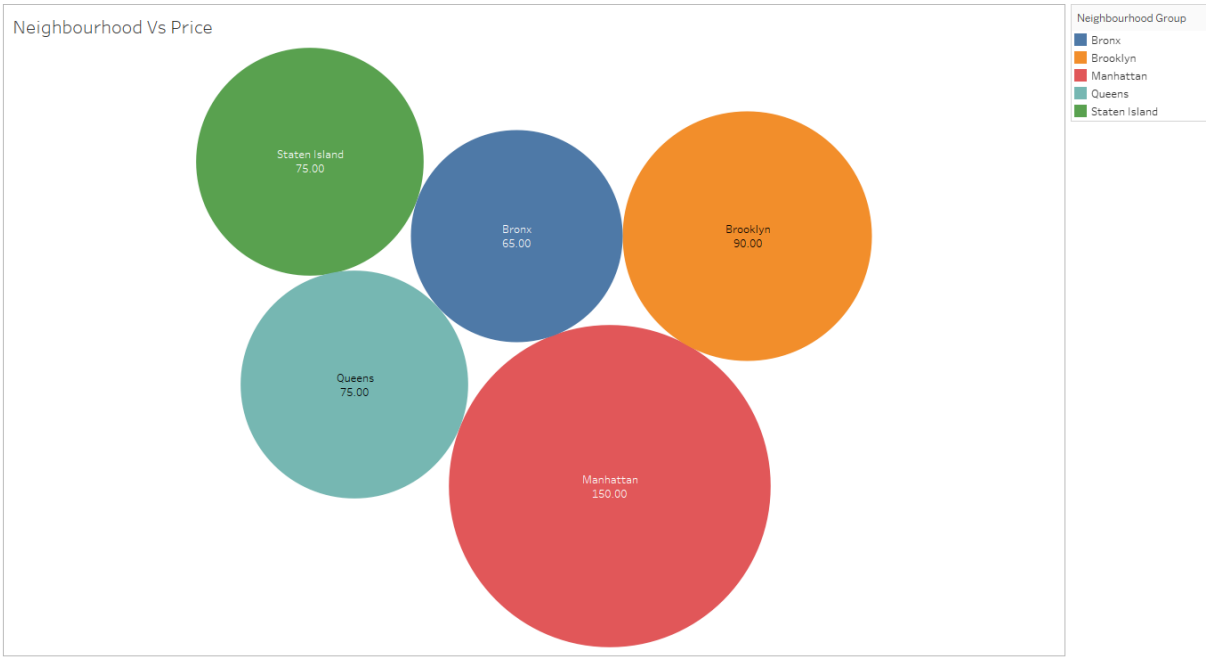
Data Analysis and Visualizations using Tableau:



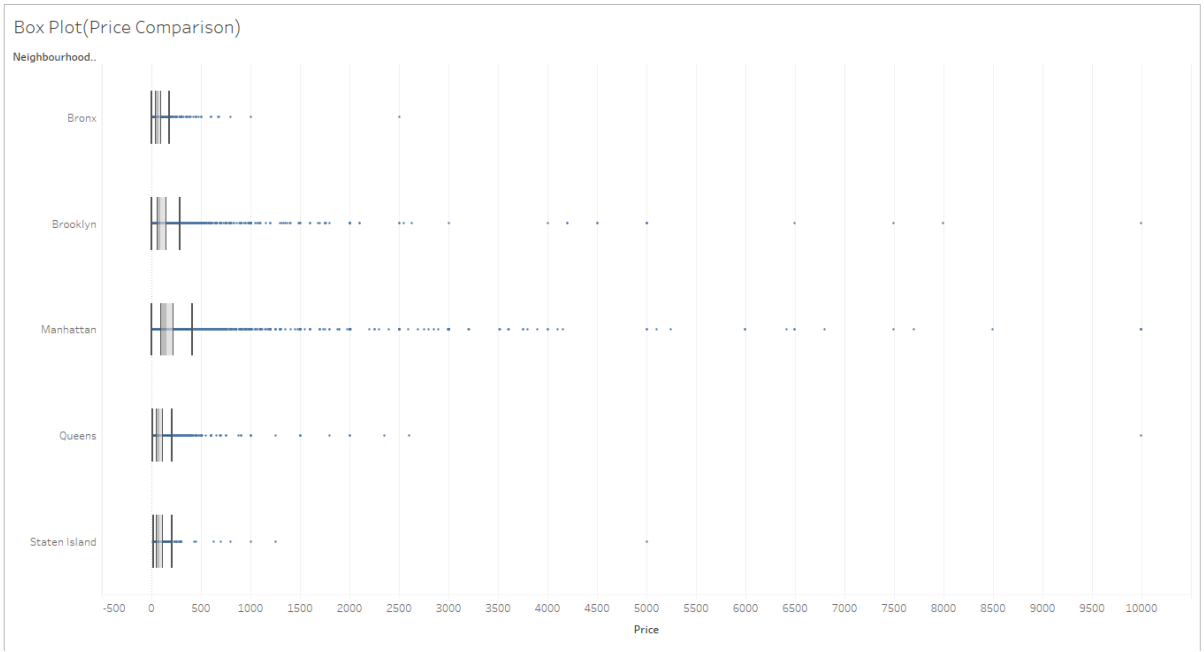
The above is the bar chart for neighbourhood groups and Manhattan has the maximum number of listings.



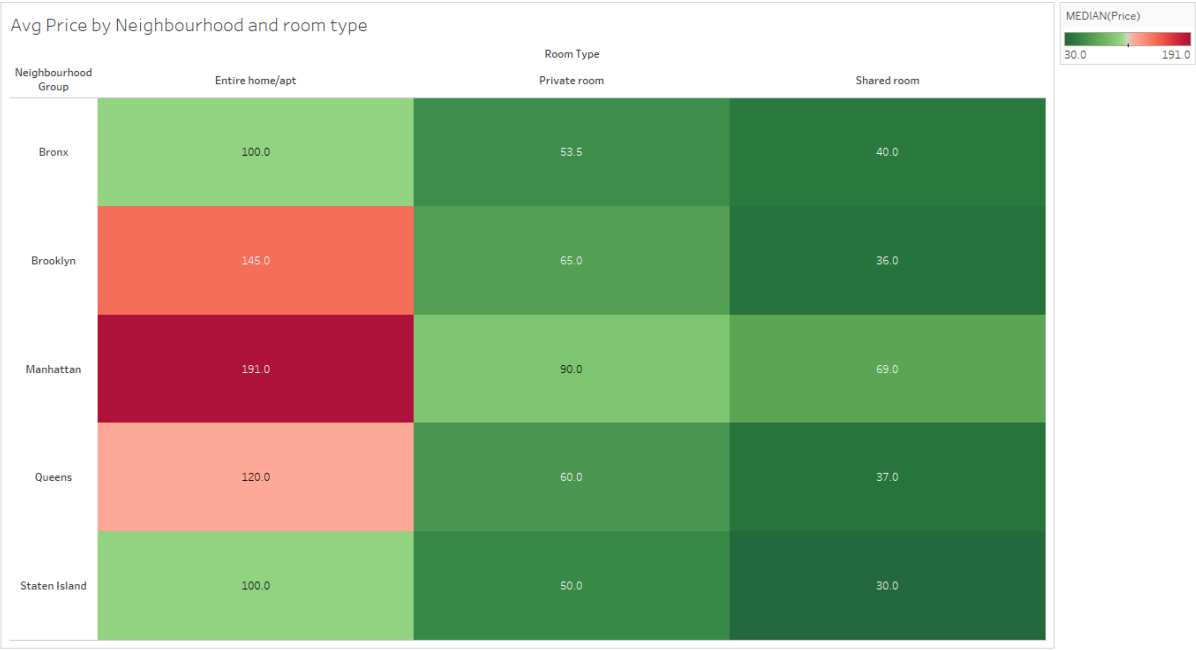
The above is a bar chart for neighbourhood groups and room types. The shared room is least preferred to that of entire home/apt and private room.



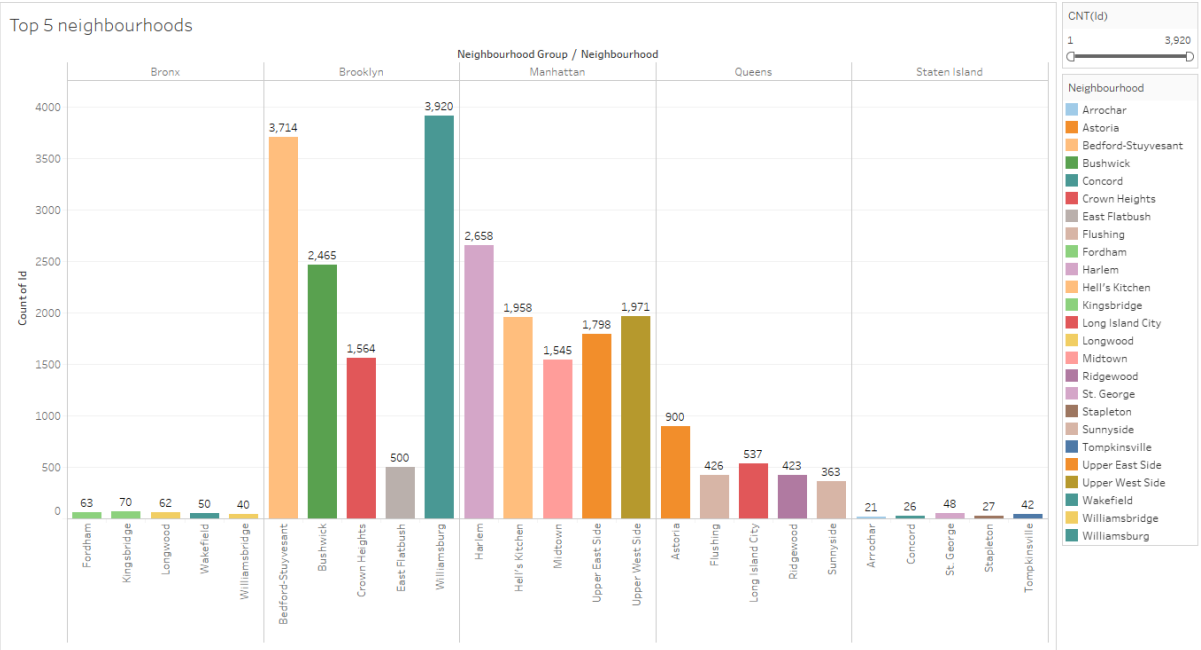
The above is a bubble chart with neighbourhood groups and median of the prices. Manhattan has the highest price.



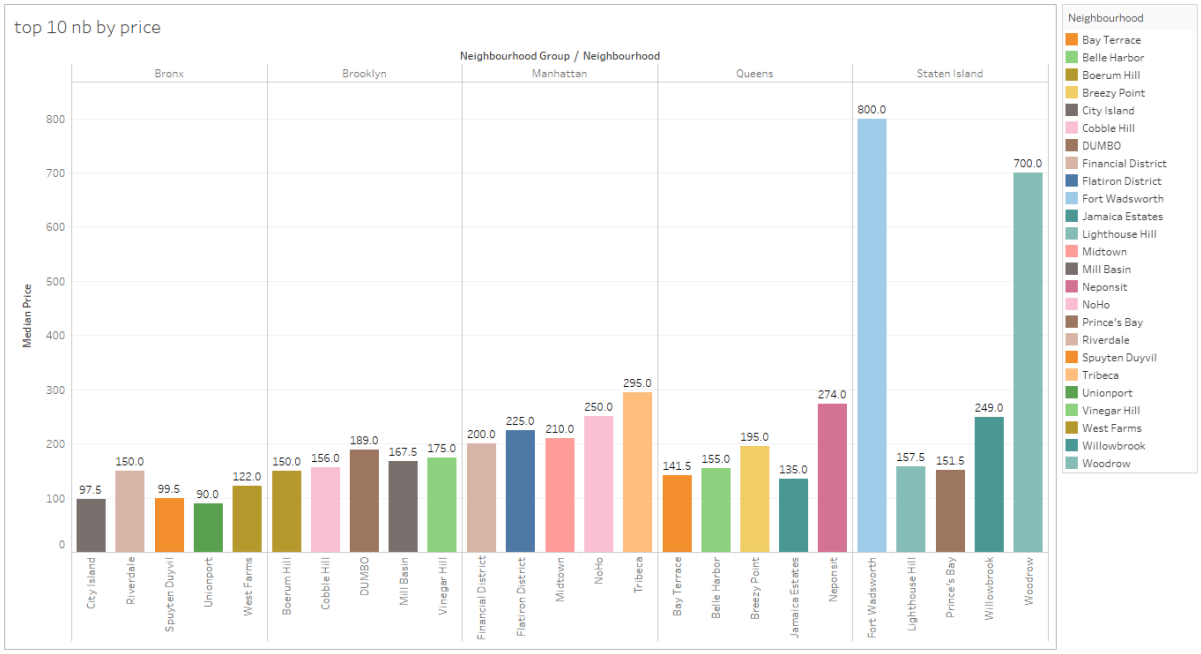
The above is box plot for prices through neighbourhood groups. The prices are very high for Manhattan.



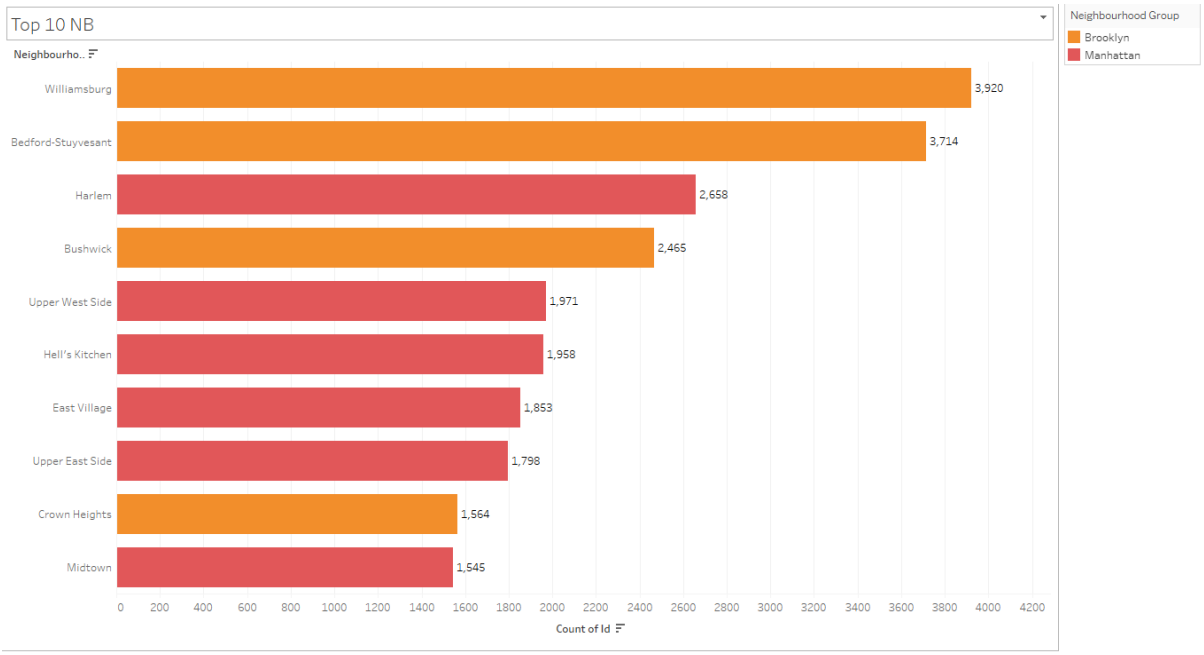
The above is a tree map for the median prices by neighbourhood group and room type. The highest prices are for entire home/apt in Manhattan.



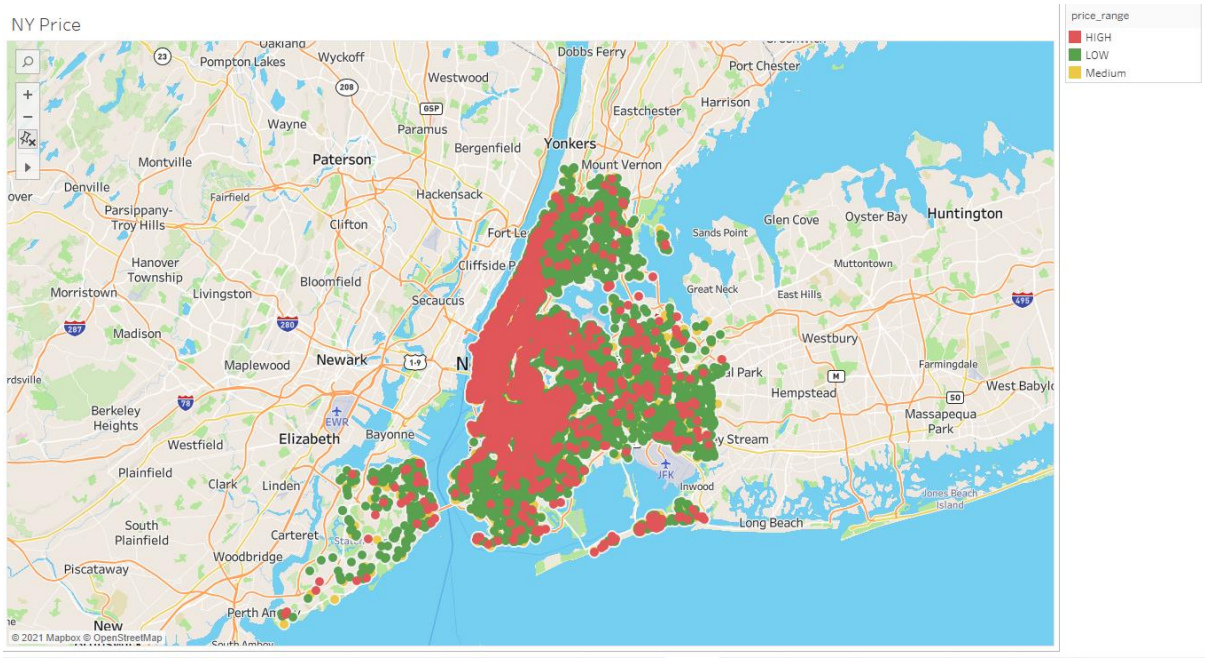
The above are the top 5 neighbourhoods of each neighbourhood group by the number of bookings.



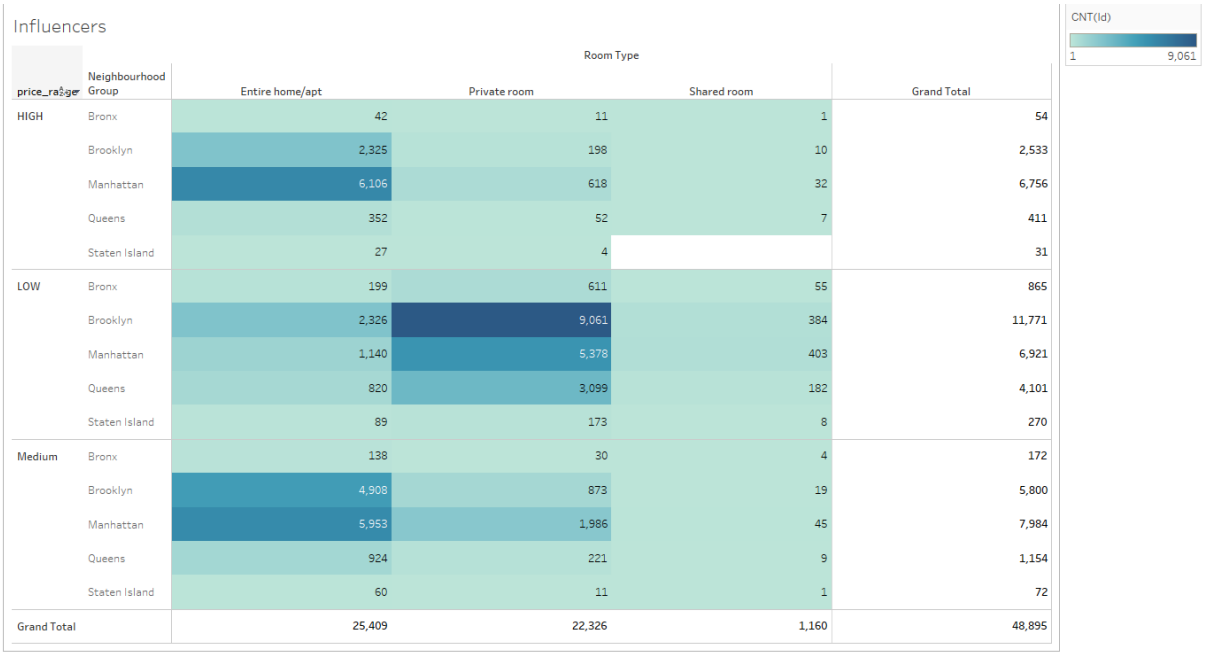
The above is bar chart for top 5 neighbourhoods of each neighbourhood group by the median of price.



The above are the top 10 neighbourhoods by bookings

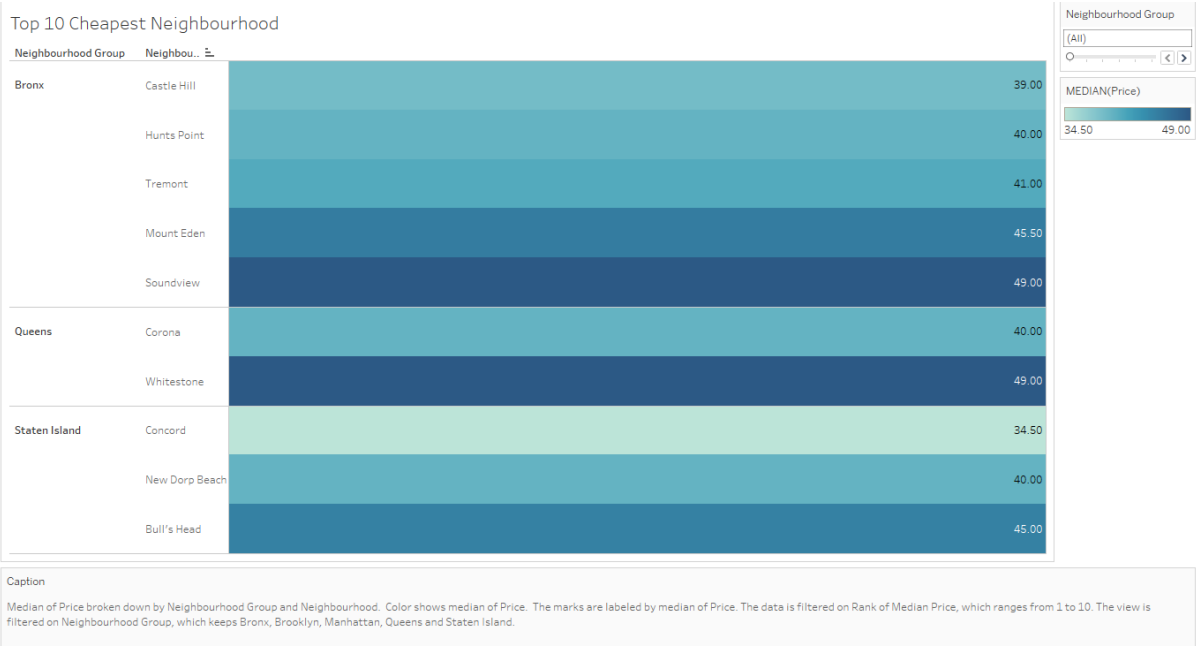


The above is a map chart for price range where a new calculated field 'price_range' is created such that if price >=200 → "HIGH", if price <=100 → LOW else MEDIUM

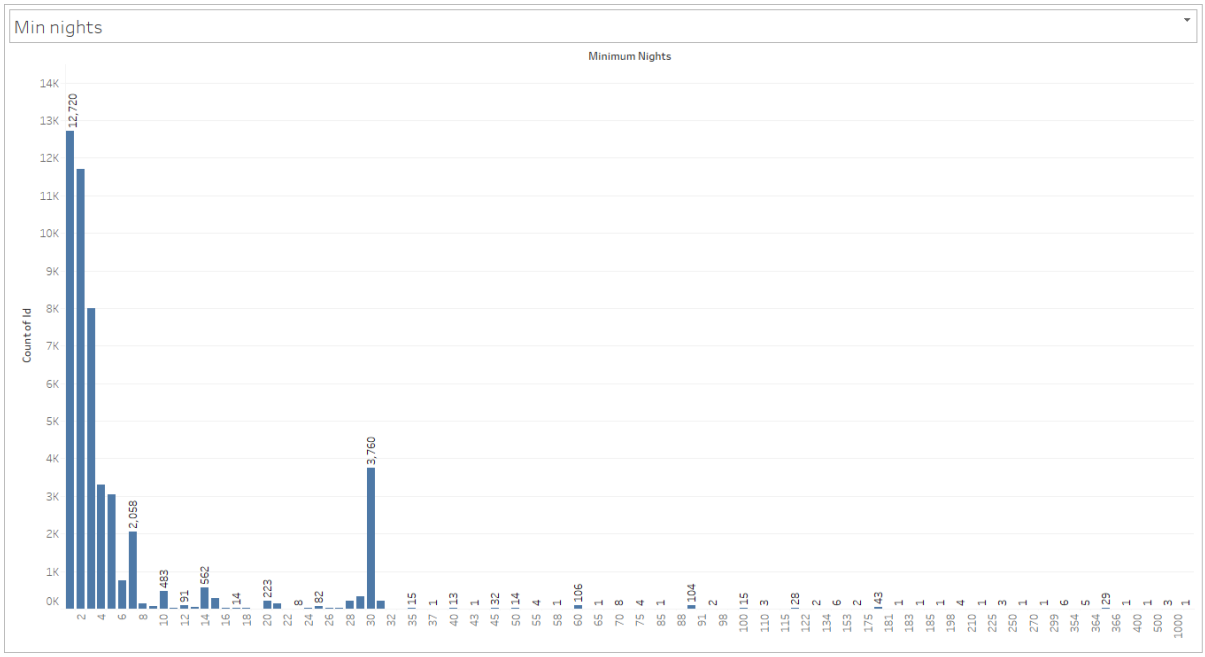


		Room Type			
price_range	Neighbourhood Group	Entire home/apt	Private room	Shared room	Grand Total
HIGH	Bronx	42	11	1	54
	Brooklyn	2,325	198	10	2,533
	Manhattan	6,106	618	32	6,756
	Queens	352	52	7	411
	Staten Island	27	4		31
LOW	Bronx	199	611	55	865
	Brooklyn	2,326	9,061	384	11,771
	Manhattan	1,140	5,378	403	6,921
	Queens	820	3,099	182	4,101
	Staten Island	89	173	8	270
Medium	Bronx	138	30	4	172
	Brooklyn	4,908	873	19	5,800
	Manhattan	5,953	1,986	45	7,984
	Queens	924	221	9	1,154
	Staten Island	60	11	1	72
Grand Total		25,409	22,326	1,160	48,895

The above is a tree map which determines influence of price range on the bookings. Entire home/apt have more bookings irrespective of price range but private and shared room with low price range have more bookings.



The above are the top 10 cheapest neighbourhoods.



The above is the number of listings with min number of nights. Most of the listings have minimum number of nights to be 1.

Avg Nights

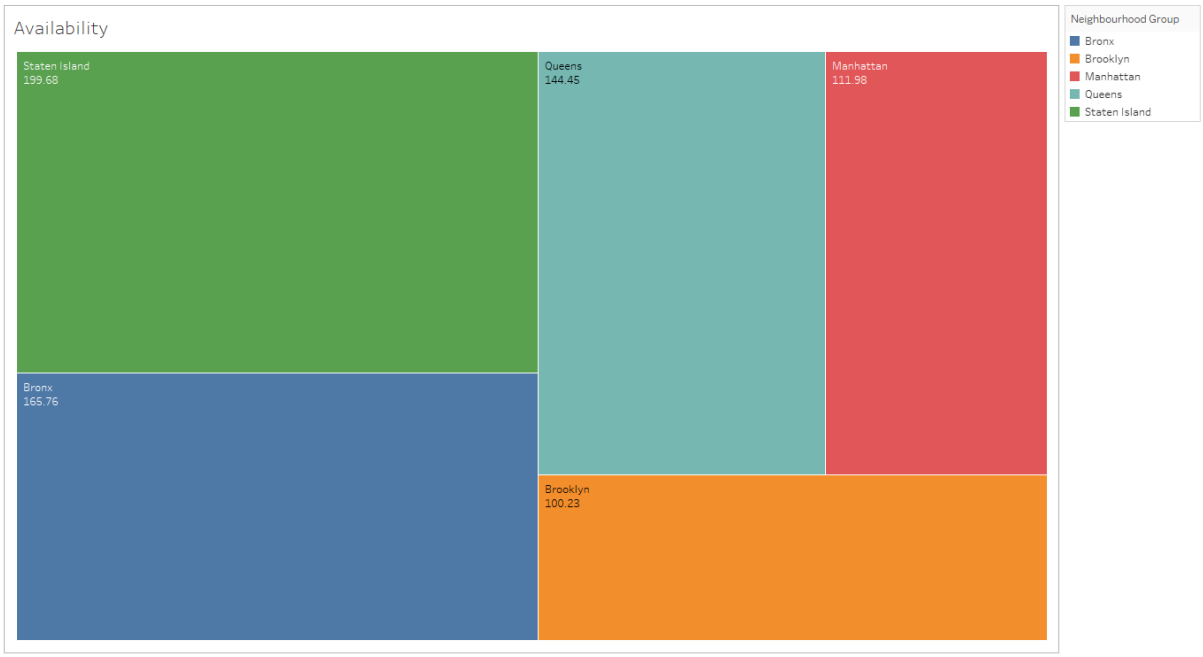
Neighbourhood Group	Avg. Minimum Nights	Count of Id	Median Minimum Nights
Manhattan	9	21,661	3
Brooklyn	6	20,104	3
Queens	5	5,666	2
Bronx	5	1,091	2
Staten Island	5	373	2

The average minimum nights are maximum for Manhattan

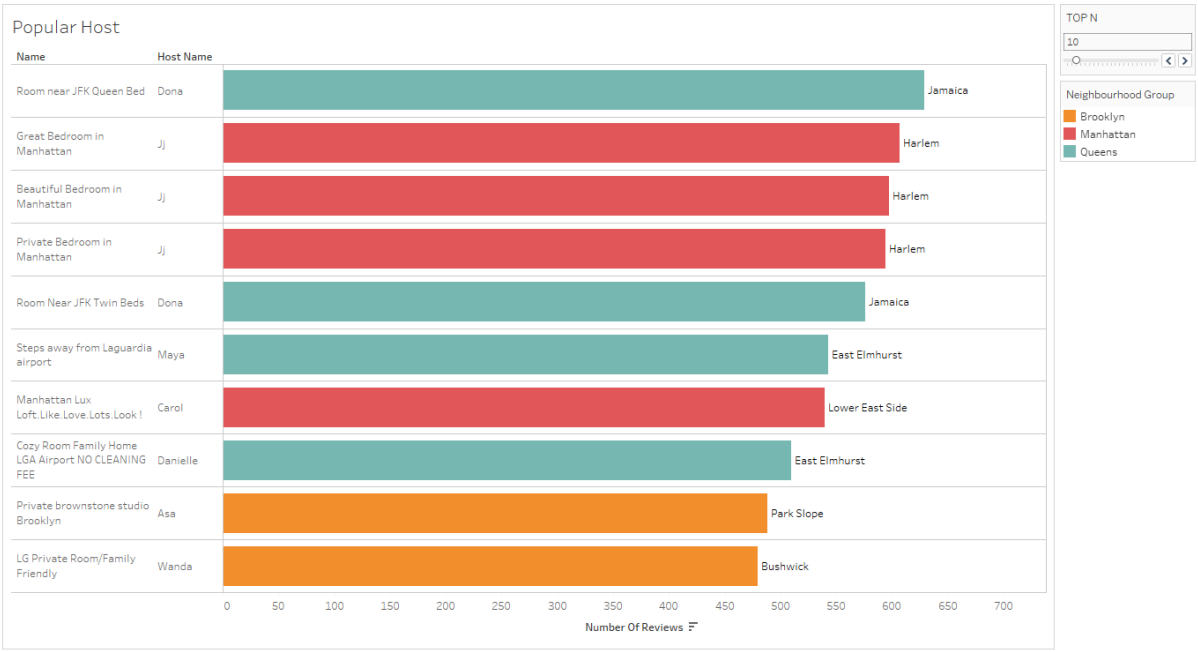
Best Performers

Host Name	Count of Id	Median Price	Price
Sonder (NYC)	327	228	82,795
Blueground	232	303	70,331
Michael	417	120	66,895
David	403	119	65,844
Alex	279	129	52,563
Jessica	205	125	50,697
John	294	99	41,892
Sally	34	165	39,789
Kara	143	239	36,723
Kevin	127	110	35,552

The above are the best performers with more price.



The above is a tree map for availability of the listings in different neighbourhood groups. The average availability is high for Staten island and least for Brooklyn.



The above is a bar chart for popular hosts based on number of reviews. Name, host name are plotted against number of reviews with neighbourhood and neighbourhood group in the colour and details.

The host Room near JFK Queen Bed (Dona) in Queens has the highest number of reviews.