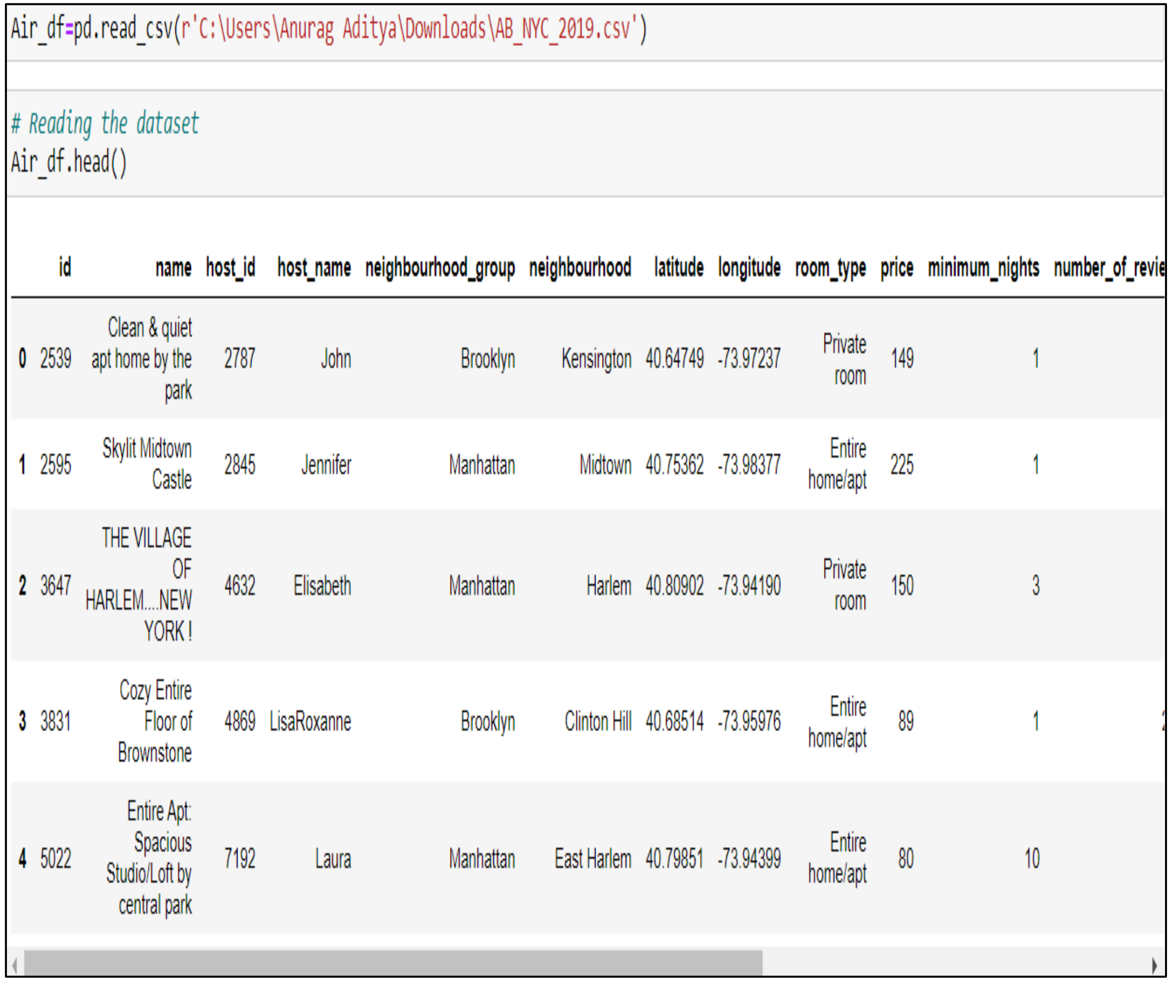
Methodology Airbnb NYC

Made By : Anurag Aditya & Pragyan Seth

Exploratory Data Analysis

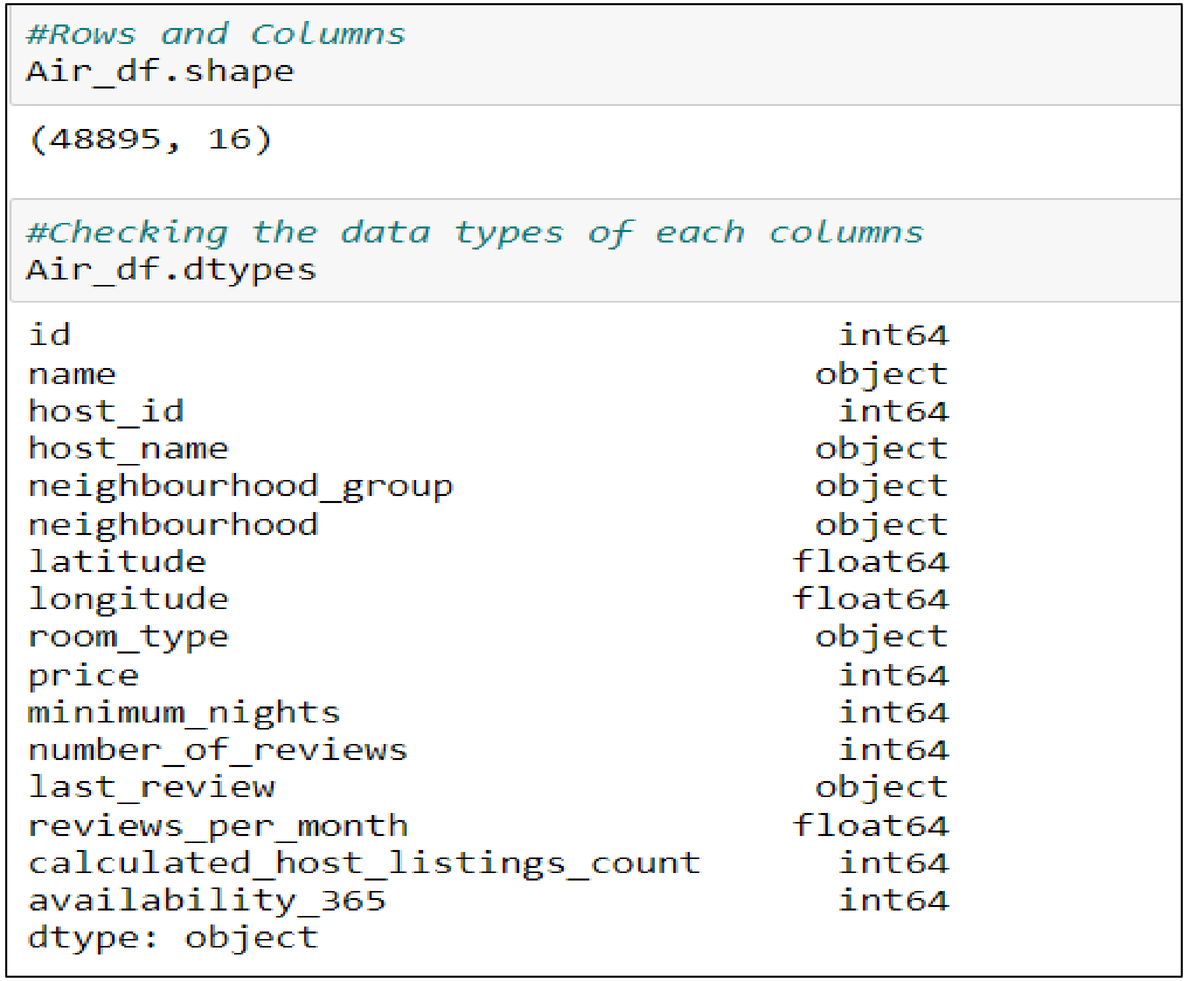
- For data analysis we have chose Python as the tool we have loaded the data set (first image) in Python and found the shape (second image) of the data set which is 48895 as rows and 16 as columns



```
Air_df=pd.read_csv(r'C:\Users\Anurag Aditya\Downloads\AB_NYC_2019.csv')
```

```
# Reading the dataset
Air_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	2
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10	



```
#Rows and Columns
Air_df.shape
```

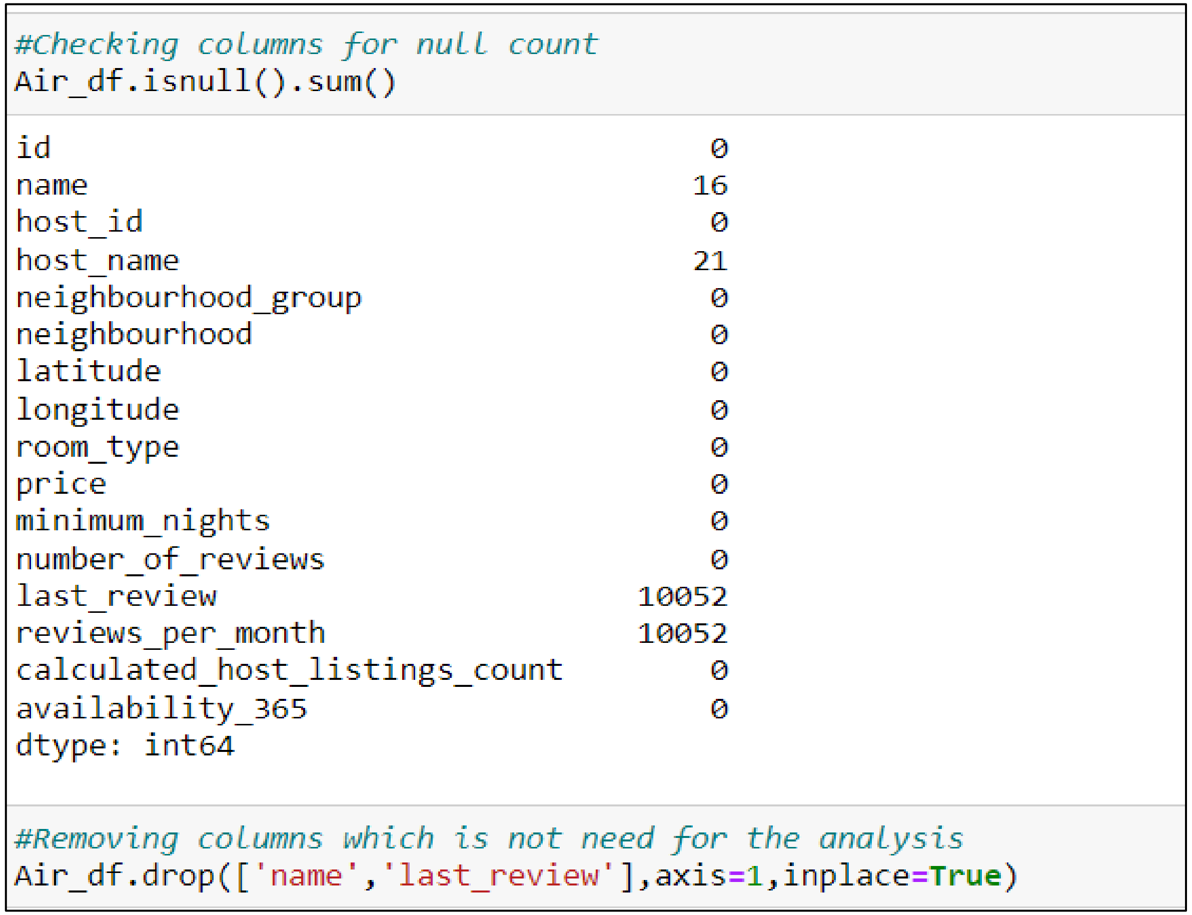
```
(48895, 16)
```

```
#Checking the data types of each columns
Air_df.dtypes
```

```
id                                  int64
name                               object
host_id                             int64
host_name                          object
neighbourhood_group                object
neighbourhood                      object
latitude                          float64
longitude                         float64
room_type                          object
price                               int64
minimum_nights                      int64
number_of_reviews                   int64
last_review                        object
reviews_per_month                 float64
calculated_host_listings_count      int64
availability_365                    int64
dtype: object
```

Exploratory Data Analysis

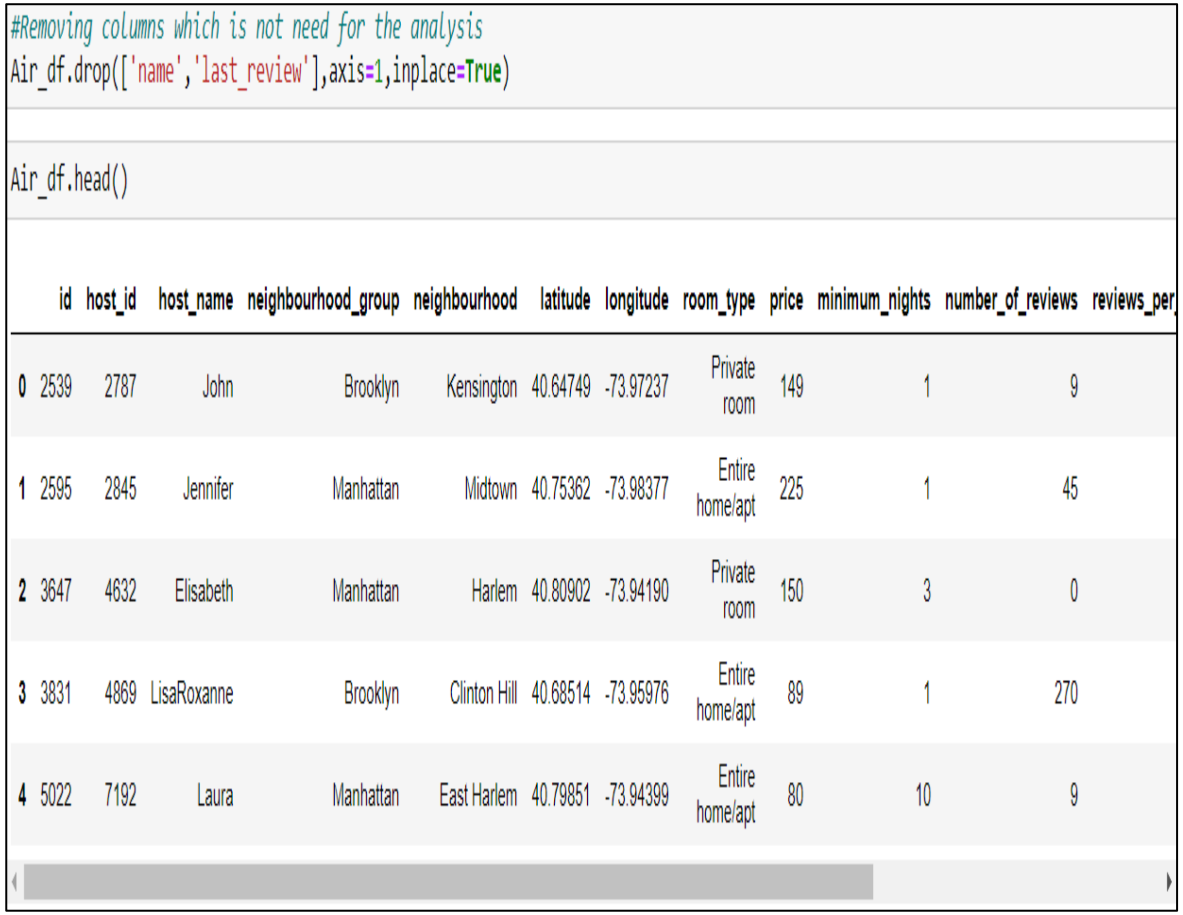
- We are checking for the null values in the columns in first image and found the null columns as (‘id’, ’host_name’, ’last_review’, ’reviews_per_month’) and on the second image we have dropped the column which we do not need for our analysis which (‘name’ , ‘last_review’)



```
#Checking columns for null count
Air_df.isnull().sum()
```

```
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
```

```
#Removing columns which is not need for the analysis
Air_df.drop(['name','last_review'],axis=1,inplace=True)
```



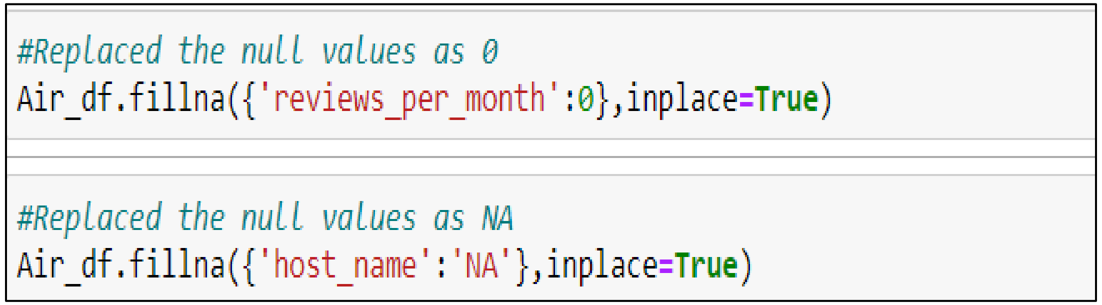
```
#Removing columns which is not need for the analysis
Air_df.drop(['name','last_review'],axis=1,inplace=True)
```

```
Air_df.head()
```

	id	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_reviews	reviews_per
0	2539	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1	9	
1	2595	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1	45	
2	3647	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3	0	
3	3831	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	1	270	
4	5022	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10	9	

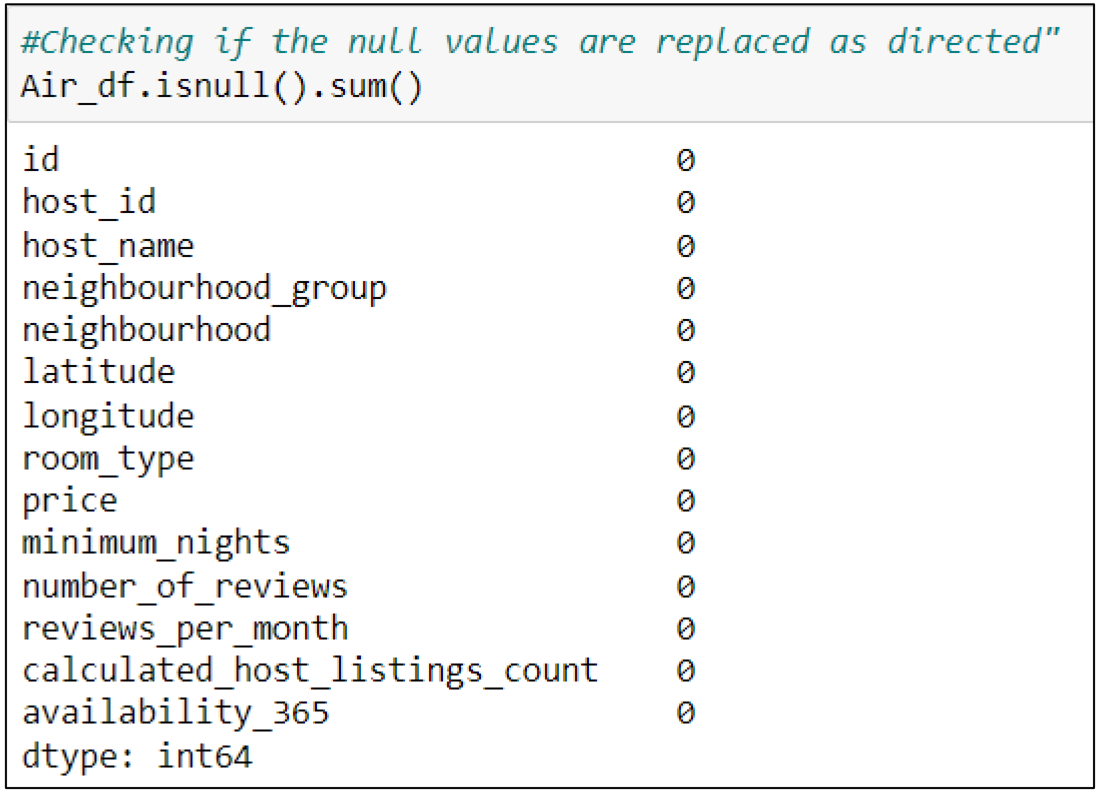
Exploratory Data Analysis

- In first Image we have replaced the null values in columns where we have found null values. In ‘review_per_month’ we have replaced it with 0 and for the column ‘host_name’ we have replaced in with ‘NA’. In second image we again checked post replacing the null values it is executed correctly.



```
#Replaced the null values as 0
Air_df.fillna({'reviews_per_month':0},inplace=True)
```

```
#Replaced the null values as NA
Air_df.fillna({'host_name':'NA'},inplace=True)
```

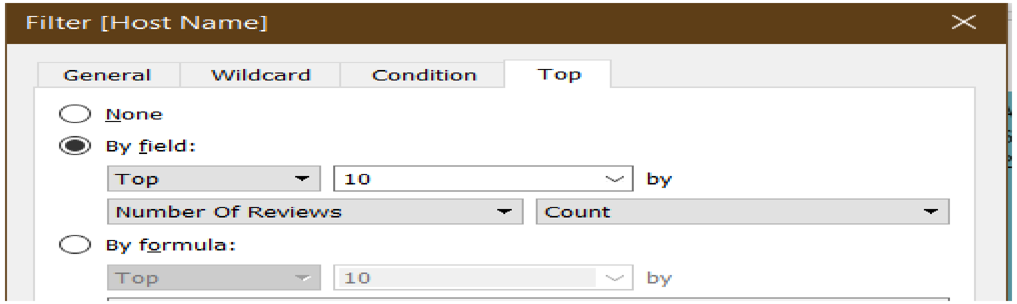


```
#Checking if the null values are replaced as directed"
Air_df.isnull().sum()
```

```
id                                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
reviews_per_month                 0
calculated_host_listings_count    0
availability_365                  0
dtype: int64
```

The visualization part is done in TABLEAU :

- For Top 10 host the analysis was done basis count of number of rows.



Filter [Host Name]

○ None

● By field:

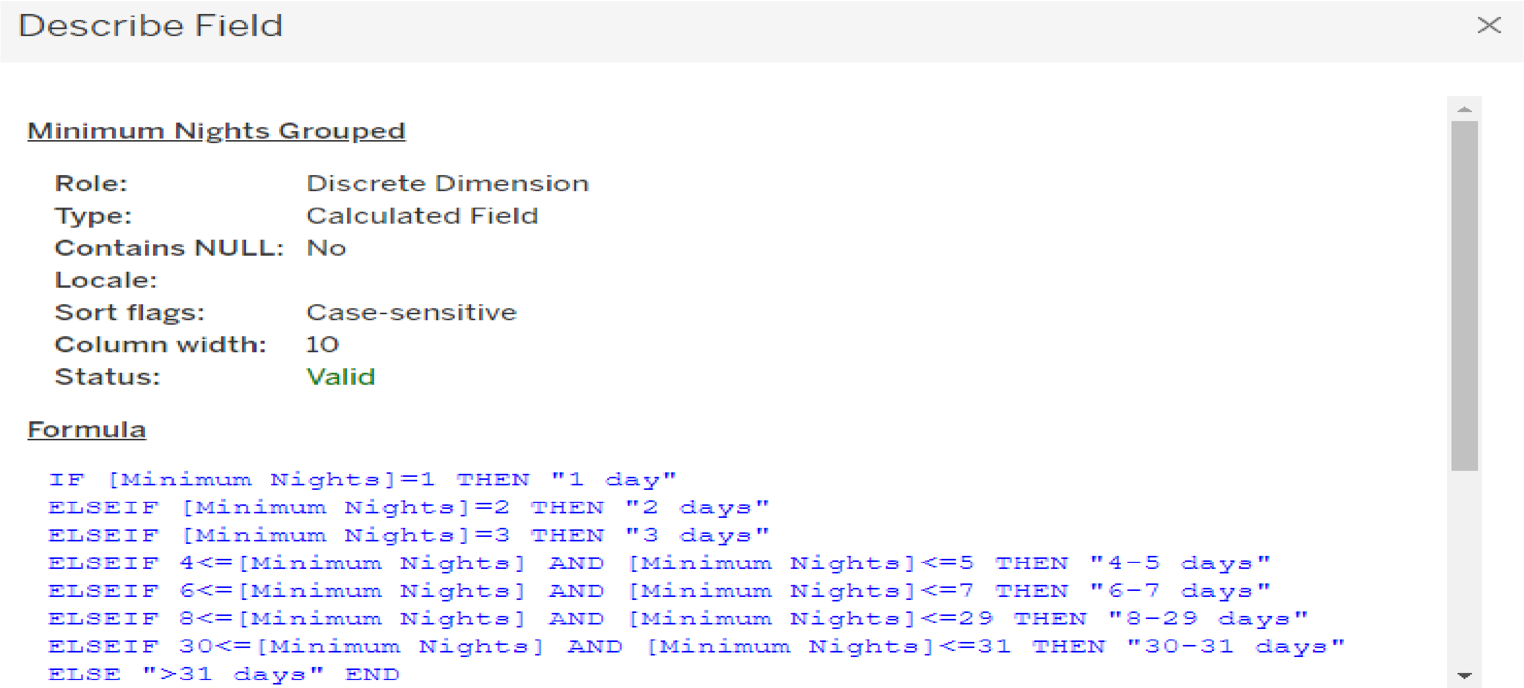
Top 10 by

Number Of Reviews Count

○ By formula:

Top 10 by

- To do the analysis on minimum nights and price bins, the bins were created



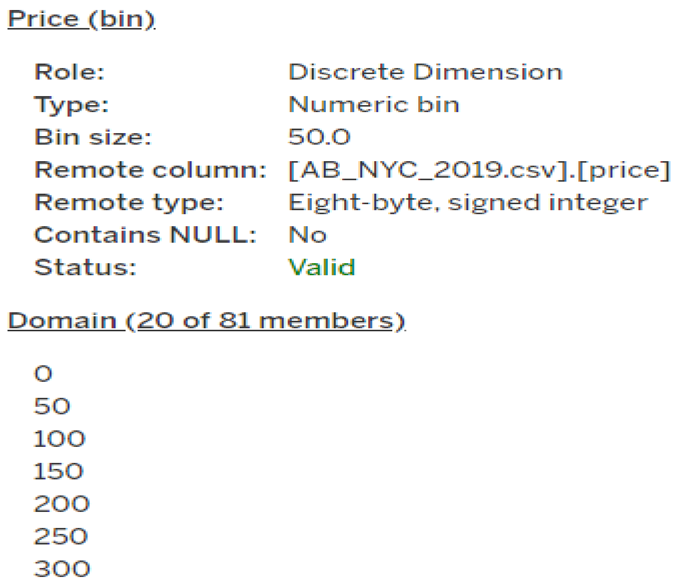
Describe Field

Minimum Nights Grouped

Role: Discrete Dimension
Type: Calculated Field
Contains NULL: No
Locale:
Sort flags: Case-sensitive
Column width: 10
Status: Valid

Formula

```
IF [Minimum Nights]=1 THEN "1 day"
ELSEIF [Minimum Nights]=2 THEN "2 days"
ELSEIF [Minimum Nights]=3 THEN "3 days"
ELSEIF 4<=[Minimum Nights] AND [Minimum Nights]<=5 THEN "4-5 days"
ELSEIF 6<=[Minimum Nights] AND [Minimum Nights]<=7 THEN "6-7 days"
ELSEIF 8<=[Minimum Nights] AND [Minimum Nights]<=29 THEN "8-29 days"
ELSEIF 30<=[Minimum Nights] AND [Minimum Nights]<=31 THEN "30-31 days"
ELSE ">31 days" END
```



Price (bin)

Role: Discrete Dimension
Type: Numeric bin
Bin size: 50.0
Remote column: [AB_NYC_2019.csv].[price]
Remote type: Eight-byte, signed integer
Contains NULL: No
Status: Valid

Domain (20 of 81 members)

0
50
100
150
200
250
300

PPT -1

1. **Airbnb listings spread in NYC**

- We have created a horizontal bars with percent of total of count of id and neighbourhood groups in colour mark card.

2. Most preferred neighborhood groups

- We created a pie chart for understanding the percentage of bookings done in neighbourhood groups.
- We added neighbourhood groups to the colours mark card and count of Id to the size

3. Room type most preferred by customer

- We created a pie chart for understanding the percentage of room type preferred w r t neighbourhood group
- We added Room Type to the colours Marks card to highlight the different Room Type in different colours and count of Id to the size

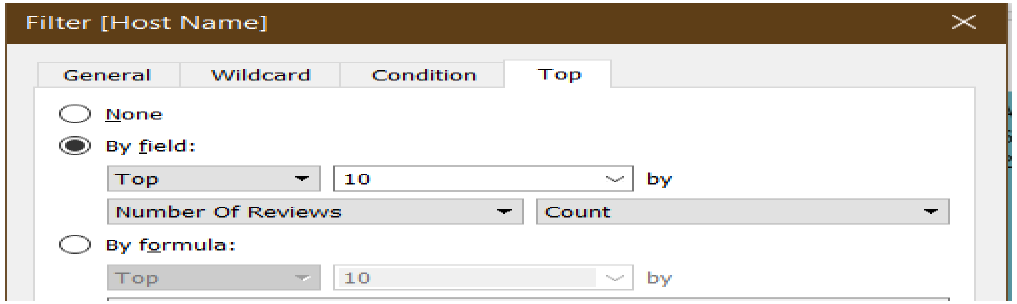
4. Average Neighbourhood group price

- We created a bubble chart with Neighbourhood Groups in Columns and Price column in Rows.
- We added the Neighbourhood Groups to the colors Marks card to highlight the different neighbourhood Groups in different colors. Also Put Avg price in Label.

5. Room type percentage basis Neighbourhood:

- We have created a side by side bars to check the percentage of booking done wrt to room type and the Neighbourhood groups. The quick table calculation was done on count of Id column kept as across

6. Top 10 hosts basis reviews:



Filter [Host Name]

○ None

● By field:

Top 10 by

Number Of Reviews Count

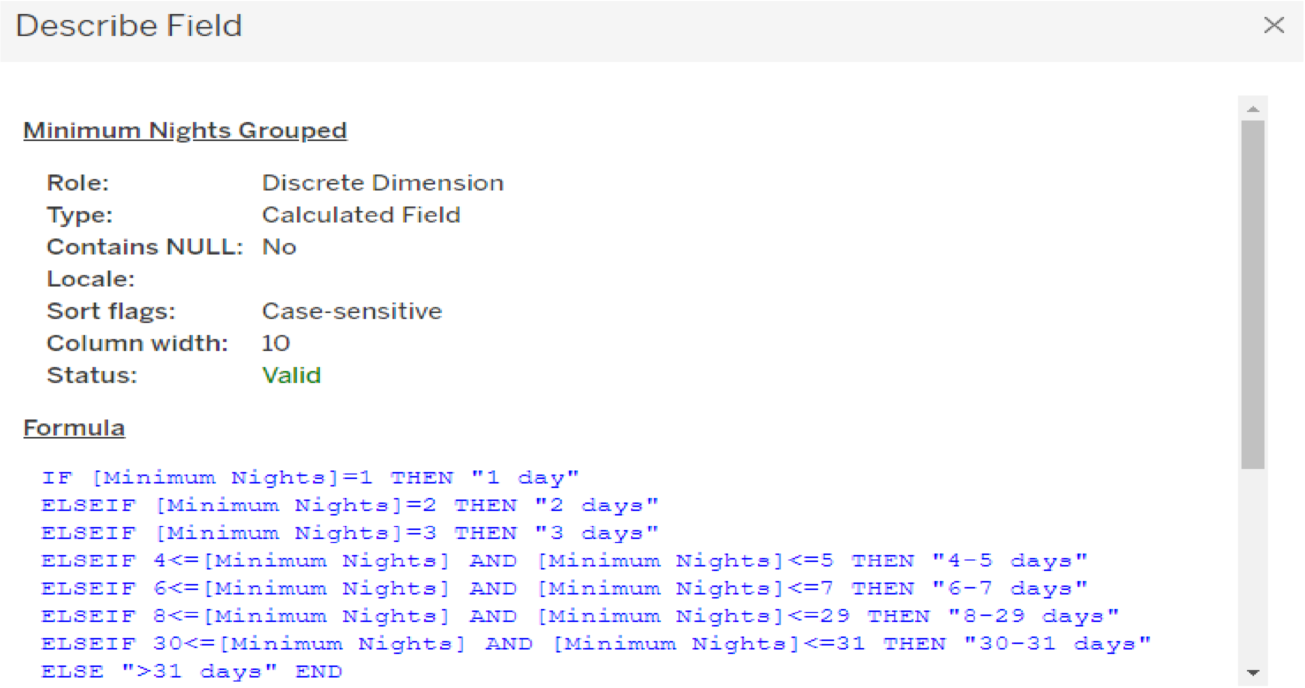
○ By formula:

Top 10 by

- **We have created a top 10 filter on Host Name basis count of number of reviews.**

7. Minimum nights booked basis room type

- We have created a bins with help of calculated field.



Minimum Nights Grouped

Role:	Discrete Dimension
Type:	Calculated Field
Contains NULL:	No
Locale:	
Sort flags:	Case-sensitive
Column width:	10
Status:	Valid

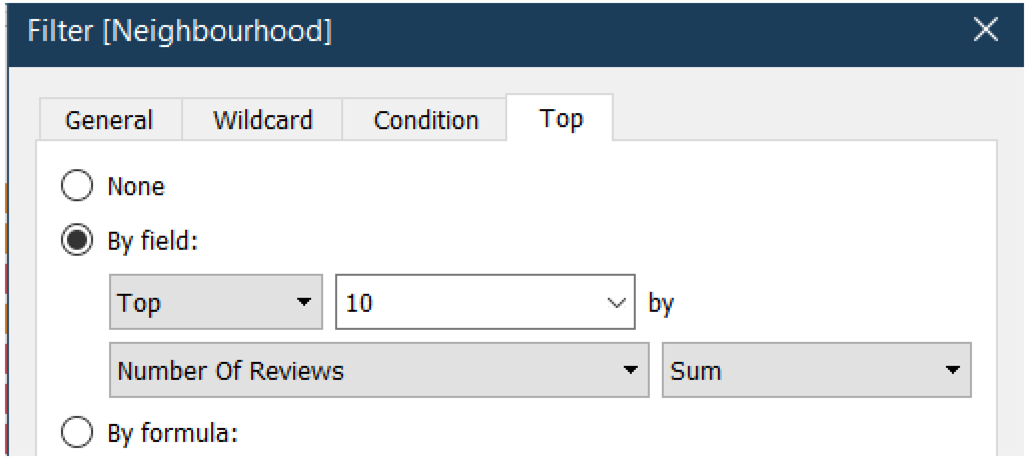
Formula

```
IF [Minimum Nights]=1 THEN "1 day"
ELSEIF [Minimum Nights]=2 THEN "2 days"
ELSEIF [Minimum Nights]=3 THEN "3 days"
ELSEIF 4<=[Minimum Nights] AND [Minimum Nights]<=5 THEN "4-5 days"
ELSEIF 6<=[Minimum Nights] AND [Minimum Nights]<=7 THEN "6-7 days"
ELSEIF 8<=[Minimum Nights] AND [Minimum Nights]<=29 THEN "8-29 days"
ELSEIF 30<=[Minimum Nights] AND [Minimum Nights]<=31 THEN "30-31 days"
ELSE ">31 days" END
```

- The bins were used to display the distribution of minimum nights based on the amount of ids booked for each neighbourhood group.

8. Popular Neighbourhood basis reviews

- We have created a top 10 filter on Neighbourhood basis sum of number of reviews and kept Neighbourhood in color marks card and sum of number of review in label marks card.



Filter [Neighbourhood]

○ None

● By field:

Top 10 by

Number Of Reviews Sum

○ By formula:

9. Neighbourhood vs Availability

- We created a dual axis chart using bar chart for availability 365 and line chart for price for top 10 neighbourhood group sorted by price and synchronized the axis accordingly.

PPT -2

1. Price variation basis geography

- We used Geo location chart to plot neighbourhood , neighbourhood Group in map to show case the variation of prices across.

2. Price range preferred by Customers

- We have created a bins of price with bin size as 50 as to analyze the price range preferred by the customer

3. Preferred Room type w.r.t Neighbourhood group

- We created Highlight Table chat by taking Room Type & Neighbourhood Group in rows and count of Id in column.

- We took the neighbourhood groups in colour Marks card to check the booking percentage of room type wrt neighbourhood groups.

4. Popular Neighborhoods:

- We took neighbourhood in rows and sum of reviews in column and took neighbourhood groups in colour.
- We used filter to show Top 10 neighbours as per the sum of reviews
- We used Geo location chart to plot neighbourhood , neighbourhood Group in map to show case the variation of prices across.

END

Thank You