DATA STORYTELLING – AIRBNB_NYC CASE STUDY

Manish Sahu, Shilpa N, Prasanna Jasawala

Objective:

To prepare for the next best steps that Airbnb needs to take as a business, we have been asked to analyse 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 following tools
- Python Jupiter Notebook
- Tableau

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

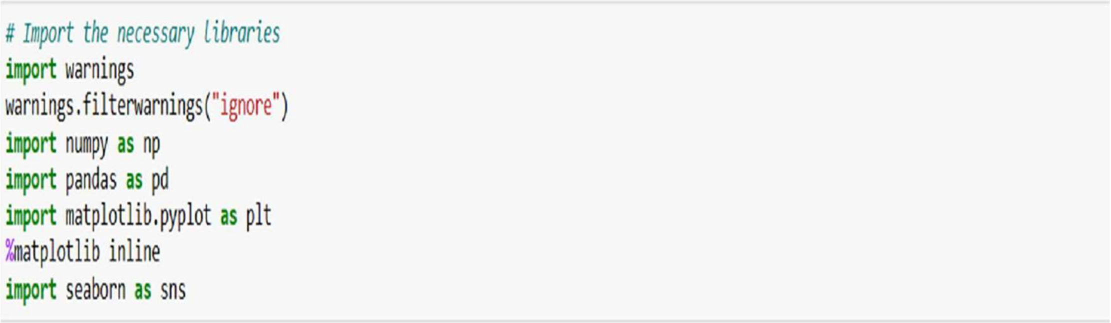
Methodology Document PPT 1:

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

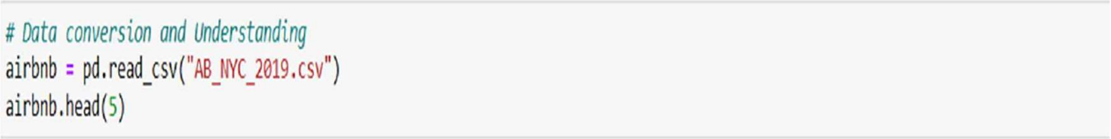
Initial Analysis using Jupiter Notebook: Data Set Used: AB_NYC_2019.csv

Number of Rows: 48895

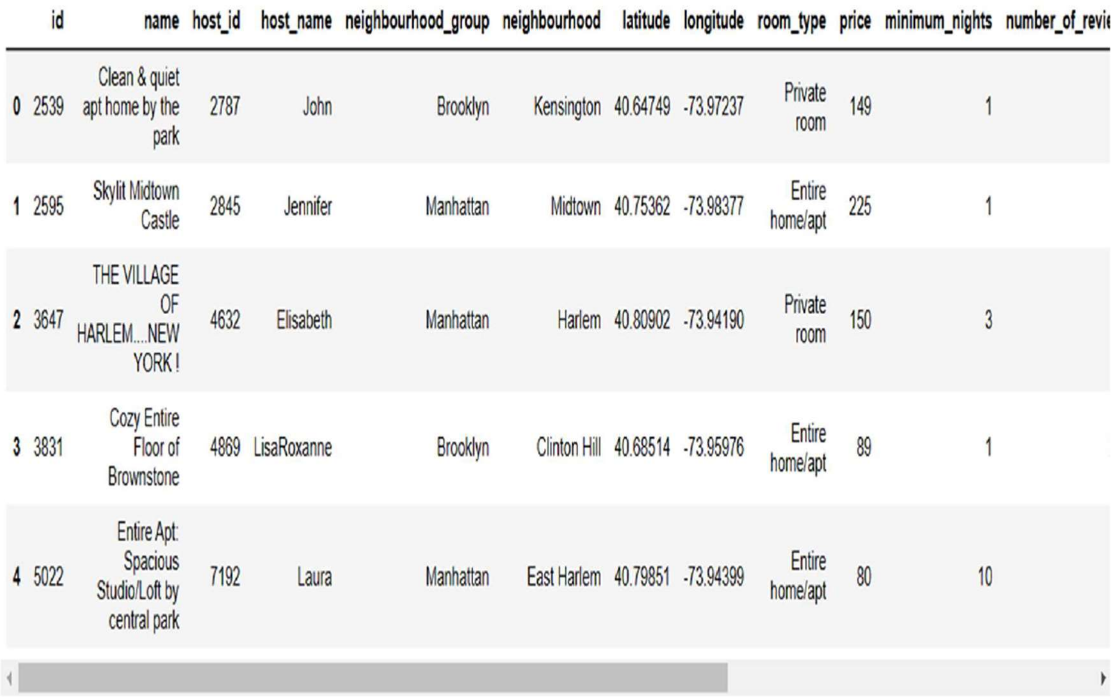
Number of Columns: 16



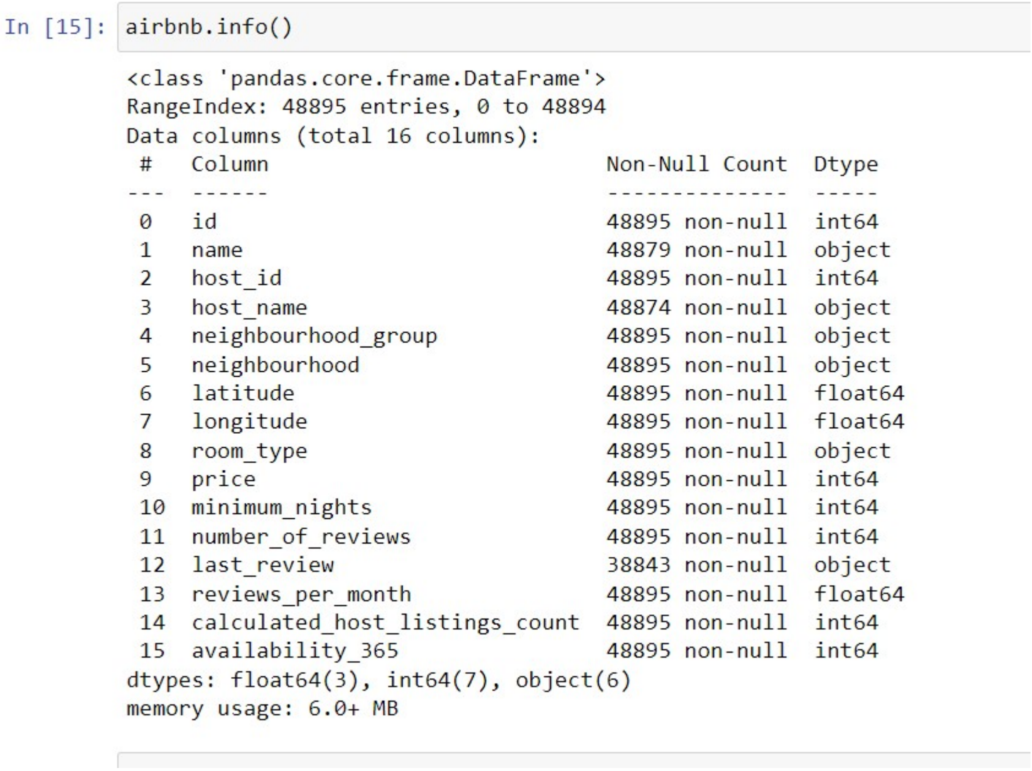
```
# Import the necessary libraries
import warnings
warnings.filterwarnings("ignore")
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```



```
# Data conversion and Understanding
airbnb = pd.read_csv("AB_NYC_2019.csv")
airbnb.head(5)
```

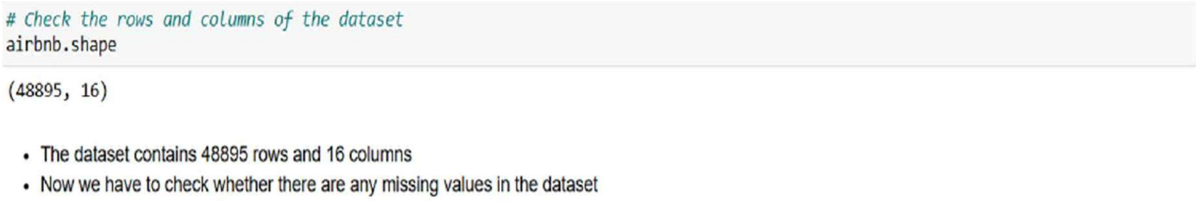


	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revie
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1	
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-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	



```
In [15]: airbnb.info()
```

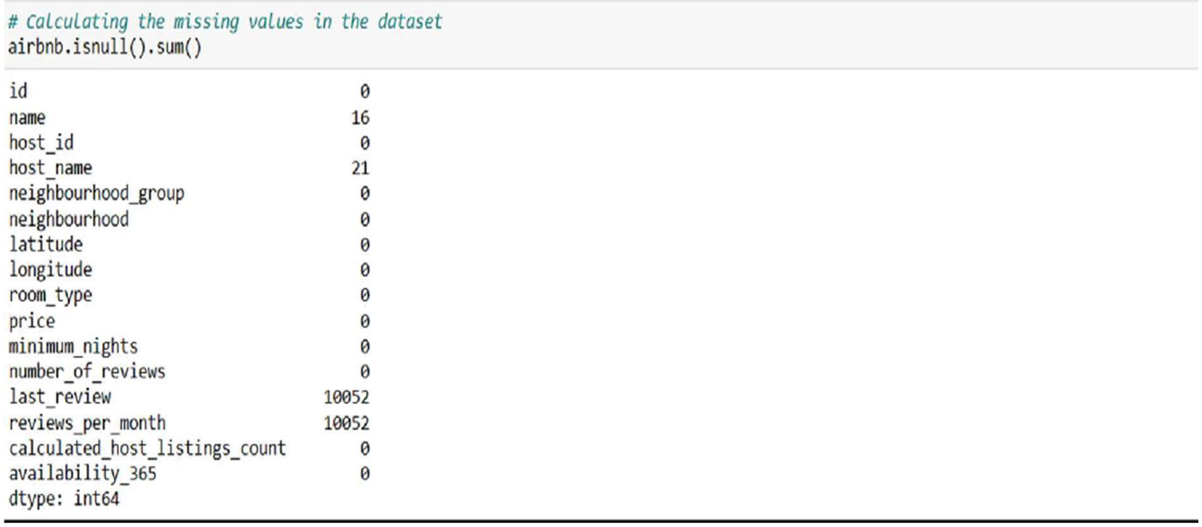
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   id                              48895 non-null  int64  
 1   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               48895 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
```



```
# Check the rows and columns of the dataset
airbnb.shape
```

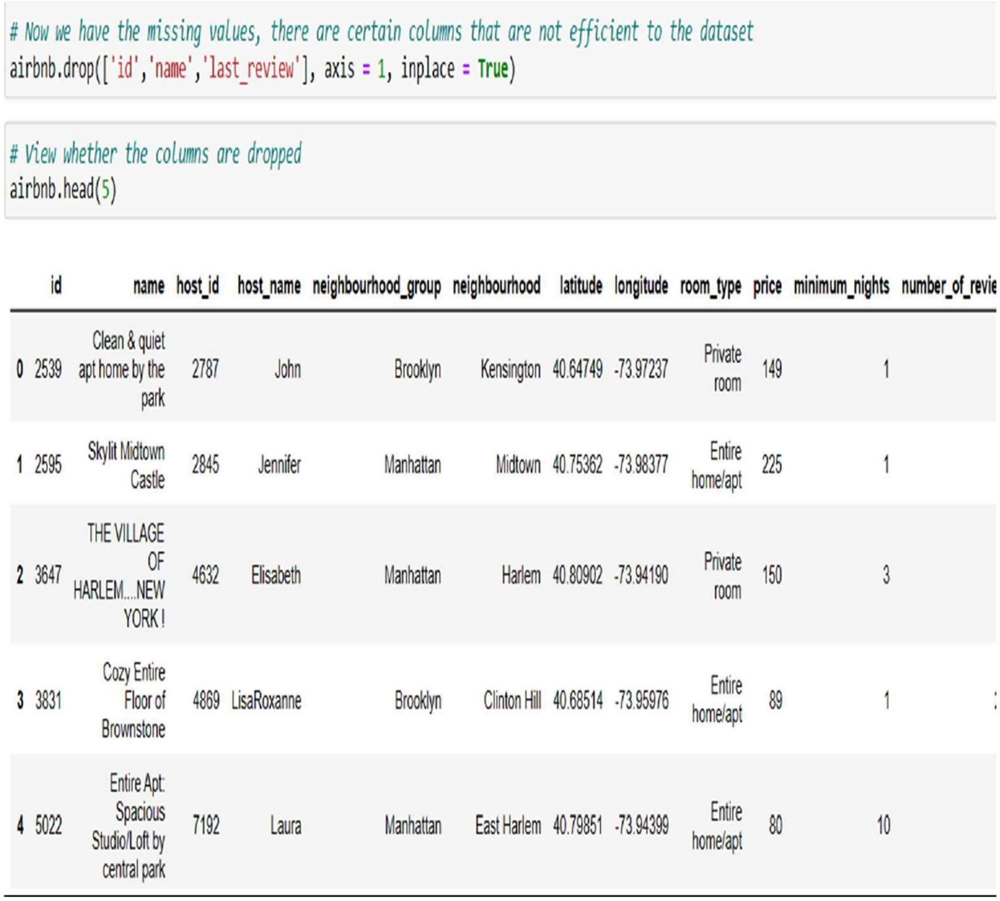
```
(48895, 16)
```

- The dataset contains 48895 rows and 16 columns
- Now we have to check whether there are any missing values in the dataset



```
# Calculating the missing values in the dataset
airbnb.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
```

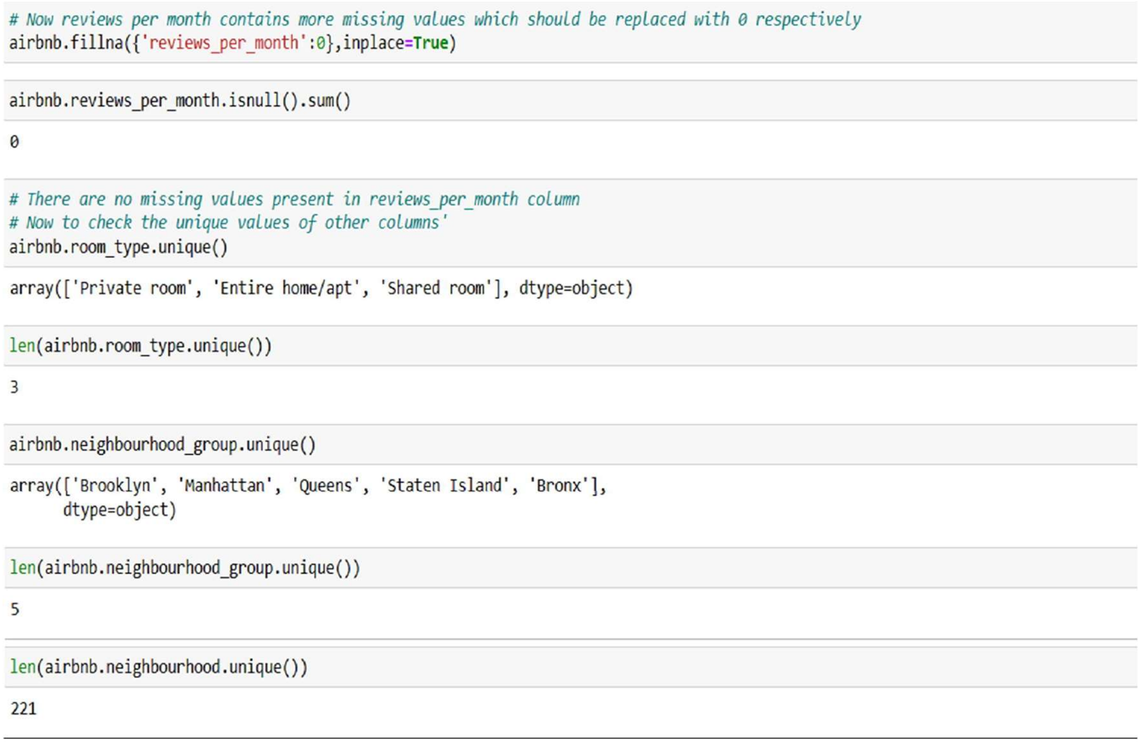


```
# Now we have the missing values, there are certain columns that are not efficient to the dataset
airbnb.drop(['id','name','last_review'], axis = 1, inplace = True)
```

```
# View whether the columns are dropped
airbnb.head(5)
```

	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	

- We removed the columns like Id, Name, Last Review which was not giving much information.



```
# Now reviews per month contains more missing values which should be replaced with 0 respectively
airbnb.fillna({'reviews_per_month':0},inplace=True)
```

```
airbnb.reviews_per_month.isnull().sum()
```

```
0
```

```
# There are no missing values present in reviews_per_month column
# Now to check the unique values of other columns'
airbnb.room_type.unique()
```

```
array(['Private room', 'Entire home/apt', 'Shared room'], dtype=object)
```

```
len(airbnb.room_type.unique())
```

```
3
```

```
airbnb.neighbourhood_group.unique()
```

```
array(['Brooklyn', 'Manhattan', 'Queens', 'Staten Island', 'Bronx'],
      dtype=object)
```

```
len(airbnb.neighbourhood_group.unique())
```

```
5
```

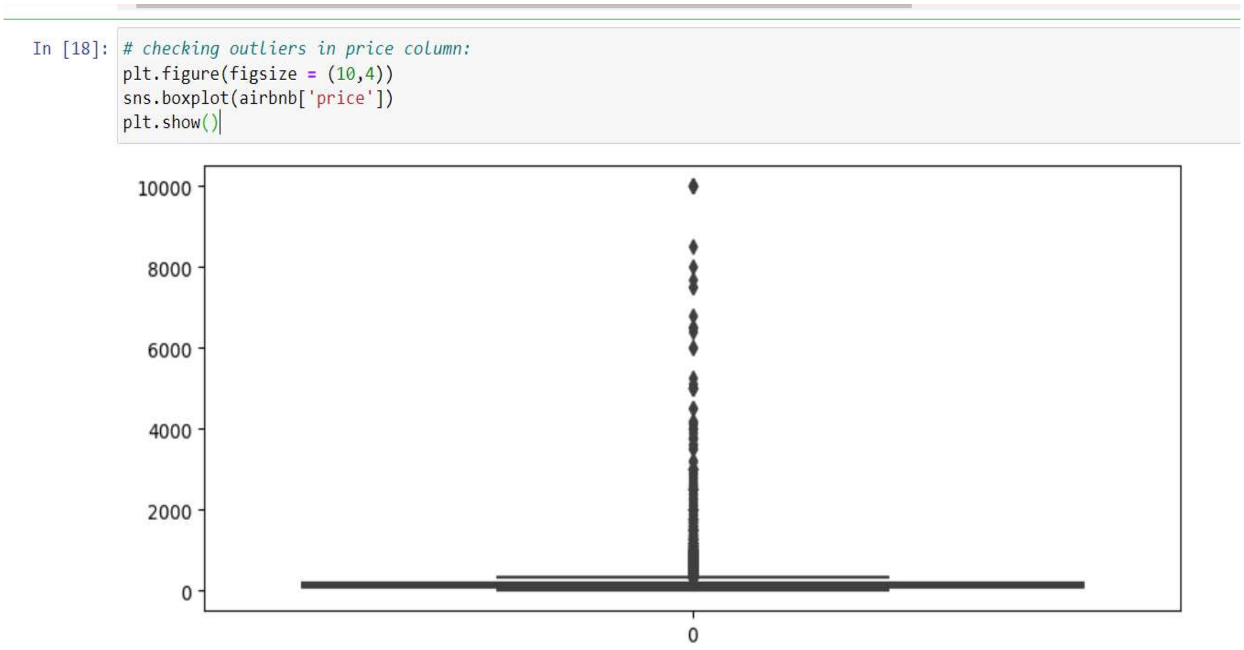
```
len(airbnb.neighbourhood.unique())
```

```
221
```

Outlier analysis and treatment:

We perform Outlier Analysis for all the numerical columns and check if required to exclude any outliers.

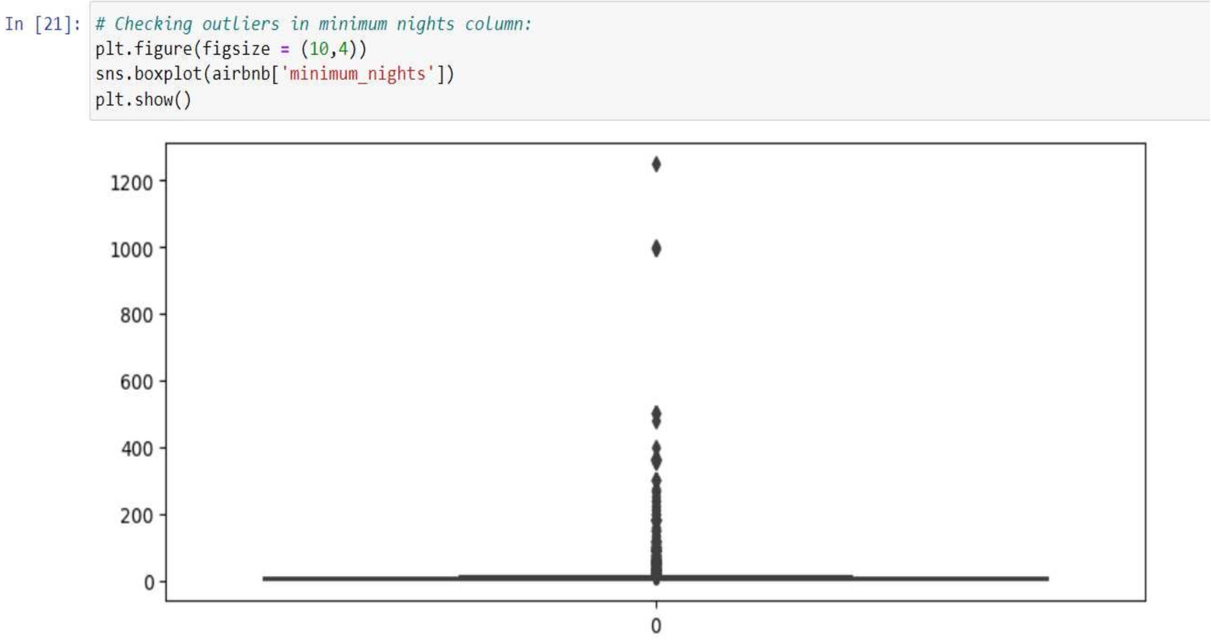
Price column analysis:



```
In [18]: # checking outliers in price column:
         plt.figure(figsize = (10,4))
         sns.boxplot(airbnb['price'])
         plt.show()
```

As we can see, there are outlier in the price column. Since we are only performing analysis, then these outlier can be useful to gain insight about the price distribution from business point.

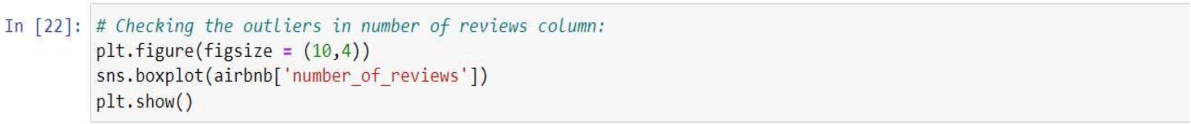
Minimum nights Analysis:



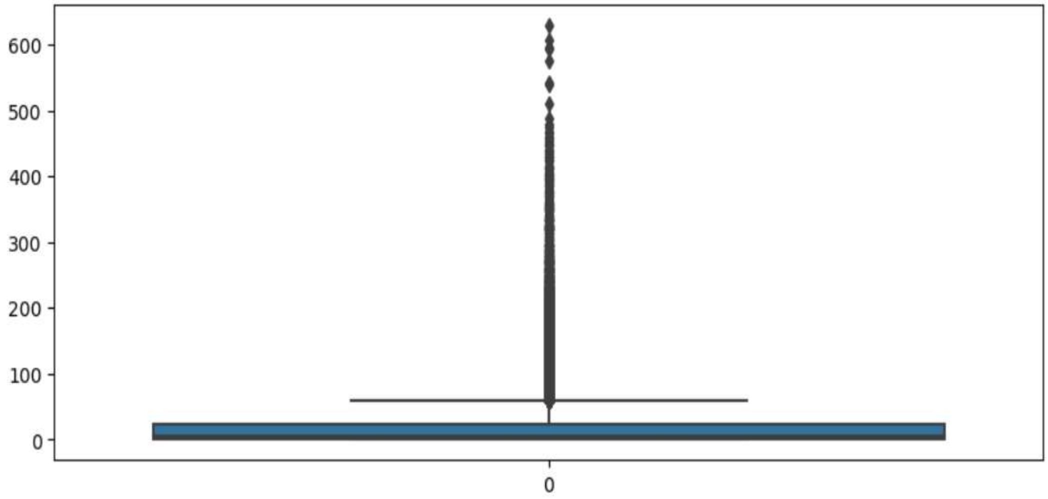
```
In [21]: # Checking outliers in minimum nights column:
         plt.figure(figsize = (10,4))
         sns.boxplot(airbnb['minimum_nights'])
         plt.show()
```

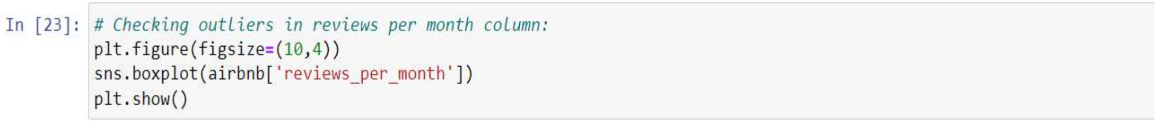
As observed, there are few outliers in Minimum Nights column as well. As these can be useful from business point of view we will keep them intact.

Reviews Analysis:

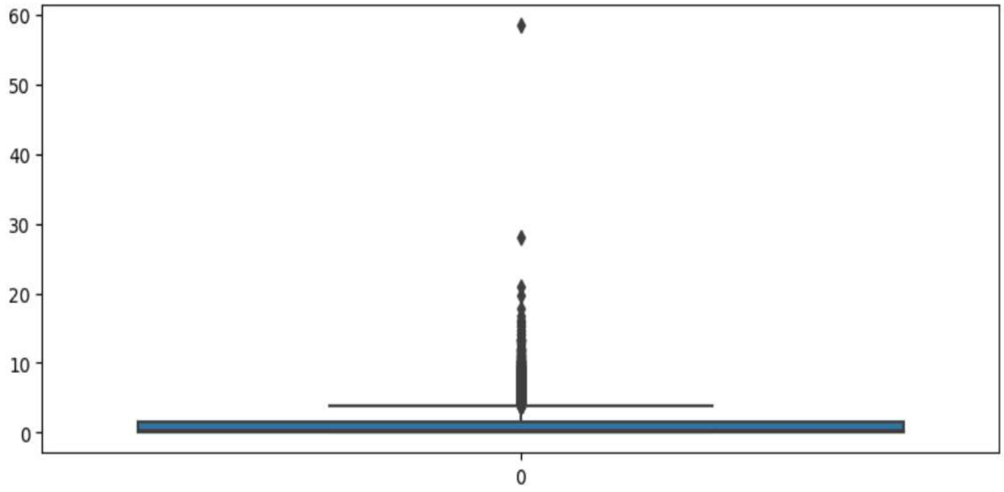


```
In [22]: # Checking the outliers in number of reviews column:
         plt.figure(figsize = (10,4))
         sns.boxplot(airbnb['number_of_reviews'])
         plt.show()
```





```
In [23]: # Checking outliers in reviews per month column:
         plt.figure(figsize=(10,4))
         sns.boxplot(airbnb['reviews_per_month'])
         plt.show()
```



The presence of outliers in the "Number of Reviews" and "Reviews per Month" columns was anticipated, as some properties are quite popular among visitors and tend to receive more reviews than others. Identifying these outliers will be helpful in our data analysis.

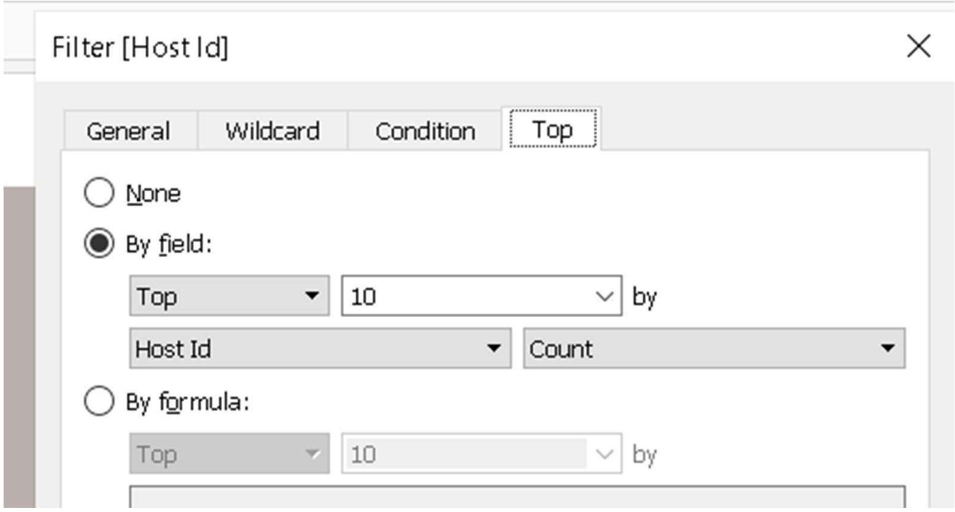
Data Wrangling:

- Checked the Duplicate rows in our dataset and no duplicate data was found.
- Checked the Null Values in our dataset. Columns like name, host-name, last review and review-per-month have null values.
- We've dropped the column name as missing values are less and dropping it won't have significant impact on analysis.
- Checked the formatting in our dataset.
- Identified and review outliers.

Data Analysis and Visualizations using Tableau:

We have used tableau to visualize the data for the assignment. Below are the detailed steps used for each visualization.

1) **Airbnb Listing Distribution in NYC:**
   - Airbnb listing distribution has been illustrated using a map chart and a bar diagram.
2) **Top 10 Host:**
   - We identified the top 10 Host Ids, Host Name with count of Host Ids using the tree map.



Filter [Host Id]

General | Wildcard | Condition | Top

○ None

● By field:

Top 10 by

Host Id Count

○ By formula:

Top 10 by

3) **Preferred Room type with respect to Neighbourhood group:**
   - We created a pie chart to understand the percentage of room types preferred in relation to different neighborhood groups.
   - We added Room Type to the colours Marks card to highlight the different Room Type in different colours and count of Host Id to the size

4) **For Variation of price with Neighbourhood Groups:**

- We used a box and whisker's plot with Neighbourhood Groups in Columns and Price in Rows.
- We changed the Price from a Sum Measure to the median measure.

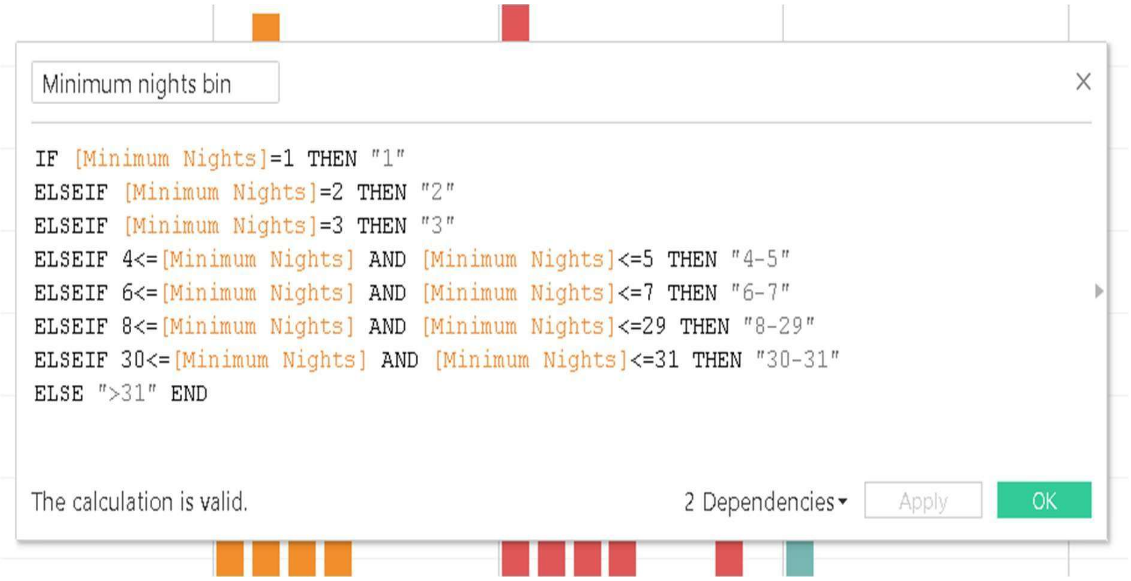
5) **Average price of Neighbourhood groups:**

• 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.

6) **Customer Booking w. r. t minimum nights:**

- We created the bin for Minimum nights as shown below.



Minimum nights bin

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

The calculation is valid.

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

7) **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 20 neighbours as per the sum of reviews.

8) **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.

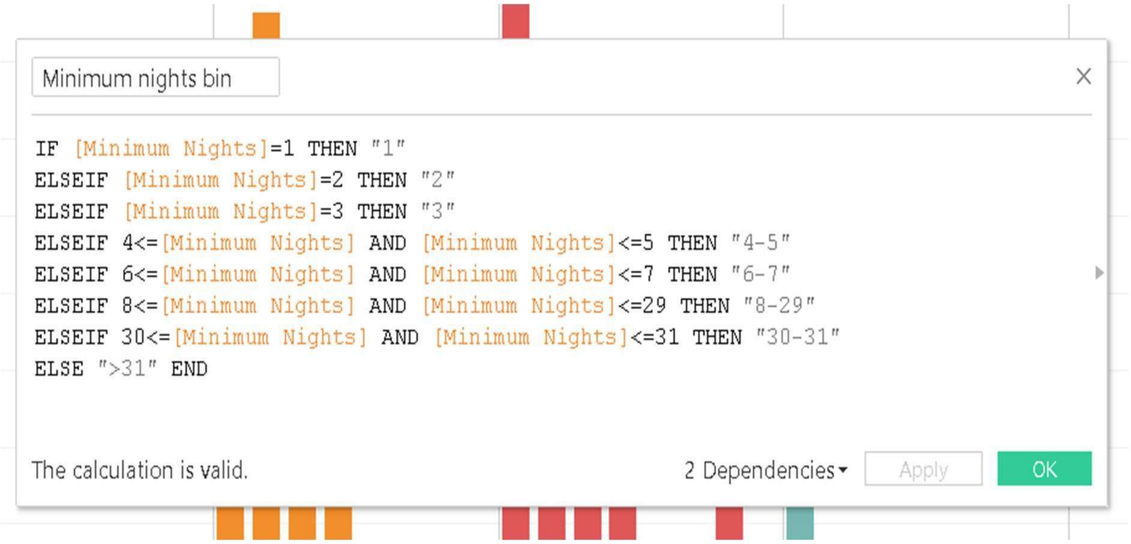
Methodology Document PPT 2:

1) **Top 10 Hosts:**
    - We identified the top 10 hosts ids, Host Name with count of Host Ids using the tree maps.

2) **Room type with respect to Neighbourhood group:**
    - 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 Host Id to the size

3) **Customer Booking with respect to minimum nights:**

- We created the bin for Minimum nights as shown below.



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

The calculation is valid.

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

4) **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.

5) **Price range preferred by Customers:**

- We have taken pricing preference based on volume of bookings done in a price range and no of Ids to create a bar chart. We have created bin for Price column with interval of $20.

6) **Understanding Price variation w.r.t Room Type & Neighbourhood:**

- We created Highlights Table chat by taking Room Type in rows & Neighbourhood Group in column.
- We took the average price in colour Marks card to highlight the different Room Type in different colours.

7) Price variation w. r. t Geography:

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

8) 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 20 neighbours as per the sum of reviews.

9) Recommendations:

  - The key recommendations have been highlighted.