AIRBNB Case Study Biplab Mondal, Indrajit Bose, Vijaya Lakshmi Badam

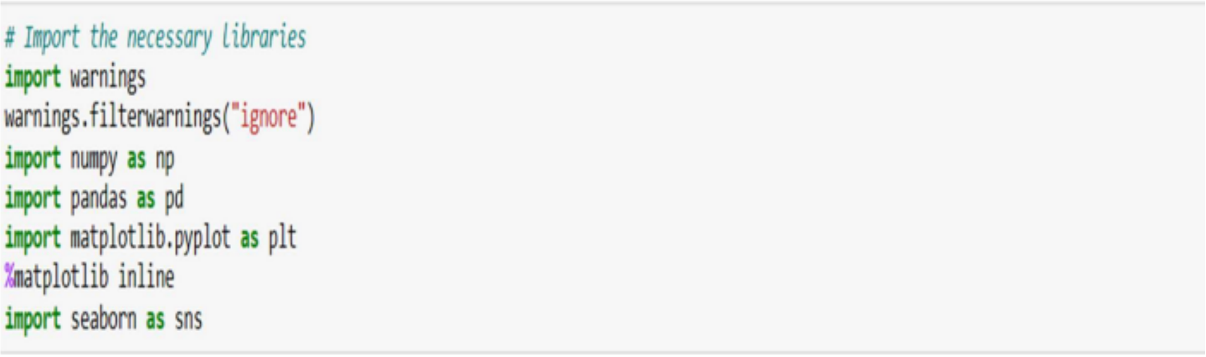
Methodology Document PPT 1:

Jupiter notebook has been used in the case study to perform initial analysis of the data and Tableau for data analysis and visualization.

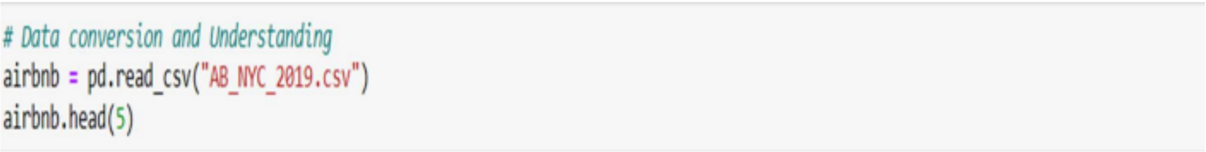
Jupiter Notebook Initial Analysis : Data Set Used: AB_NYC_2019.csv

Number of Rows: 48895

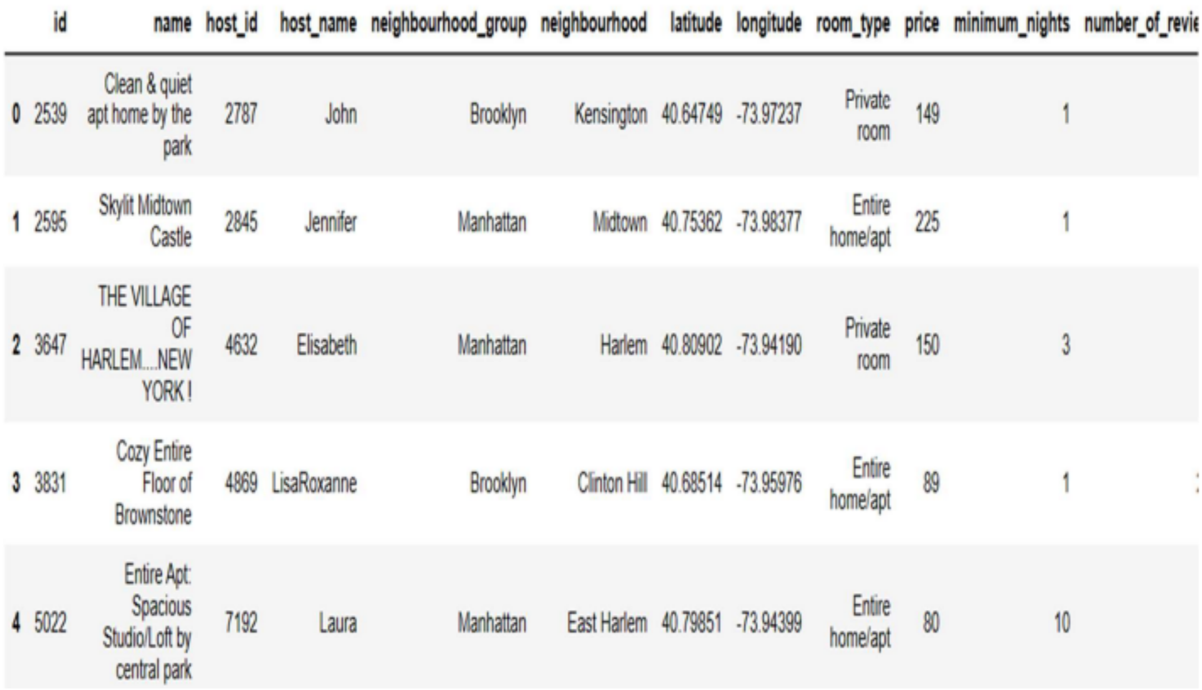
Number of Columns: 16



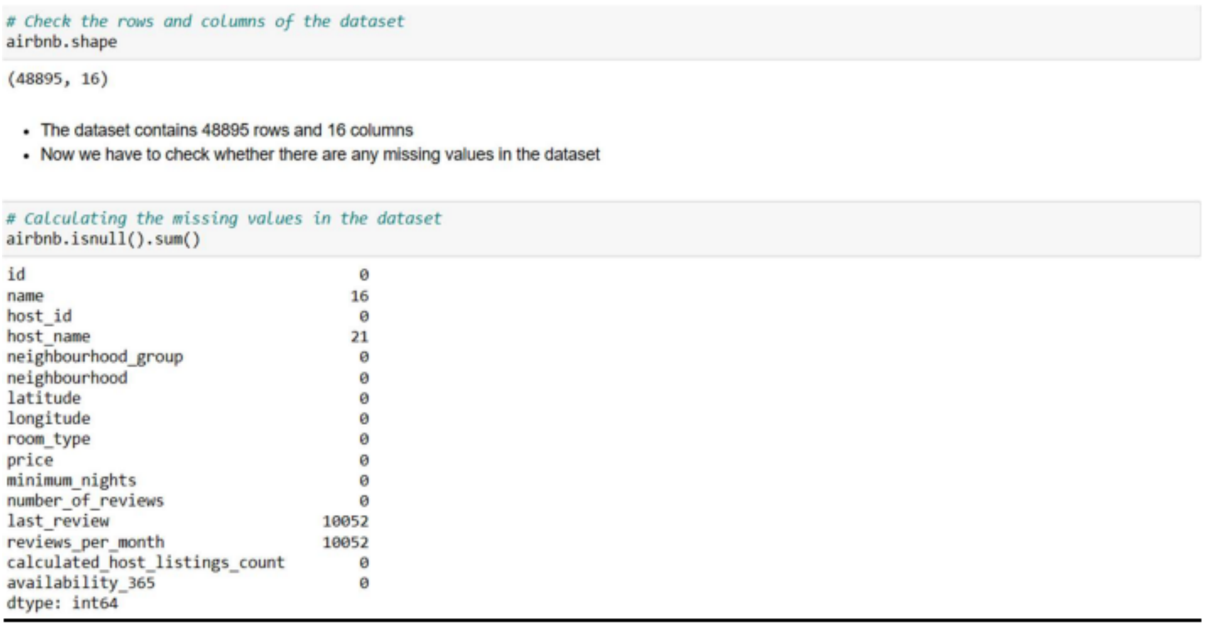
```
# Import the necessary libraries
import warnings
warnings.filterwarnings("ignore")
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```



```
# 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	



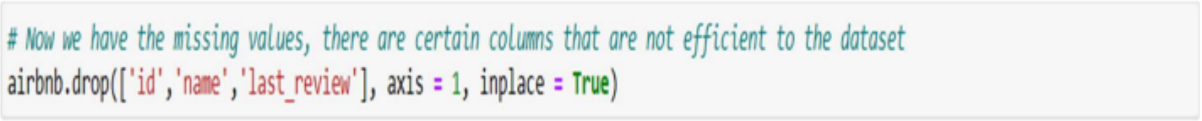
```
# Check the rows and columns of the dataset
airbnb.shape
```

```
(48895, 16)
```

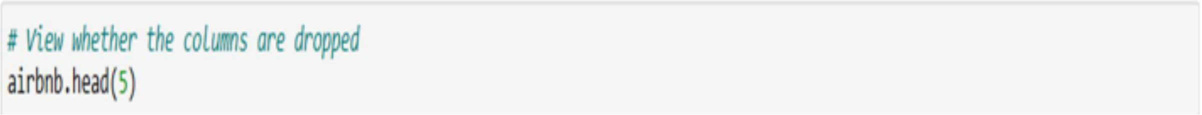
- The dataset contains 48895 rows and 16 columns
- Now we have to check whether there are any missing values in the dataset

```
# Calculating the missing values in the dataset
airbnb.isnull().sum()
```

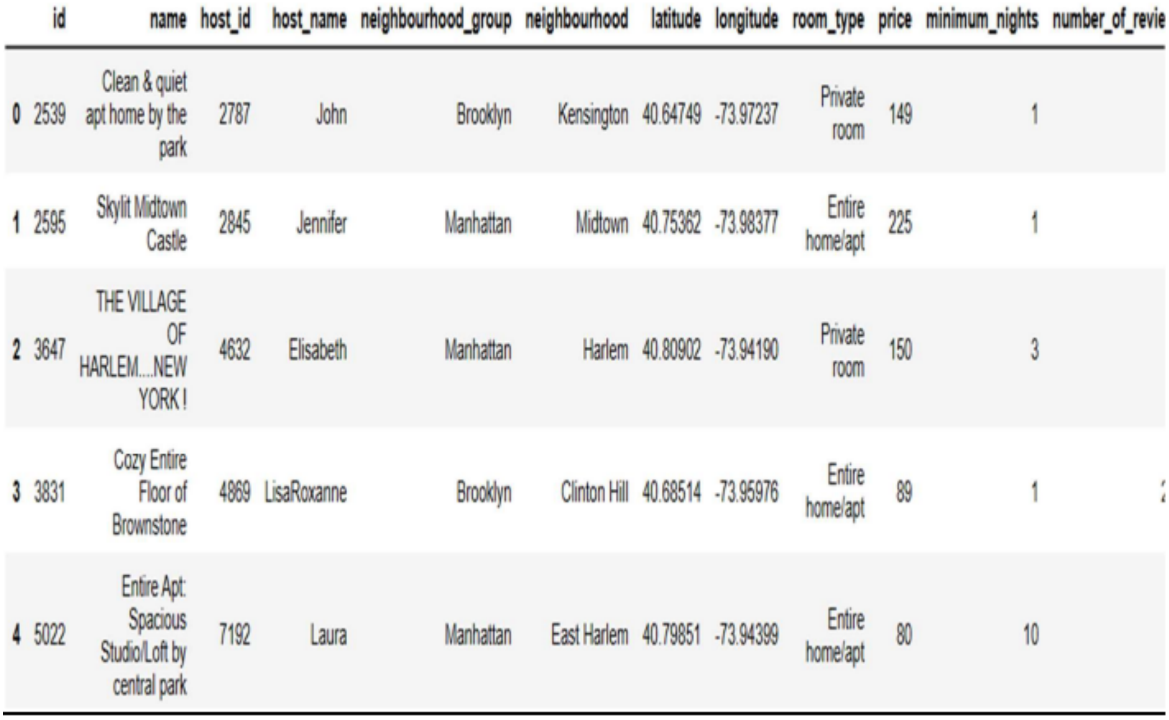
```
id                                    0
name                                 16
host_id                               0
host_name                            21
neighbourhood_group                   0
neighbourhood                         0
latitude                              0
longitude                             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	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	

Columns like Id, Name, Last Review have been removed as they were 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
```

Step 2: Data Wrangling:

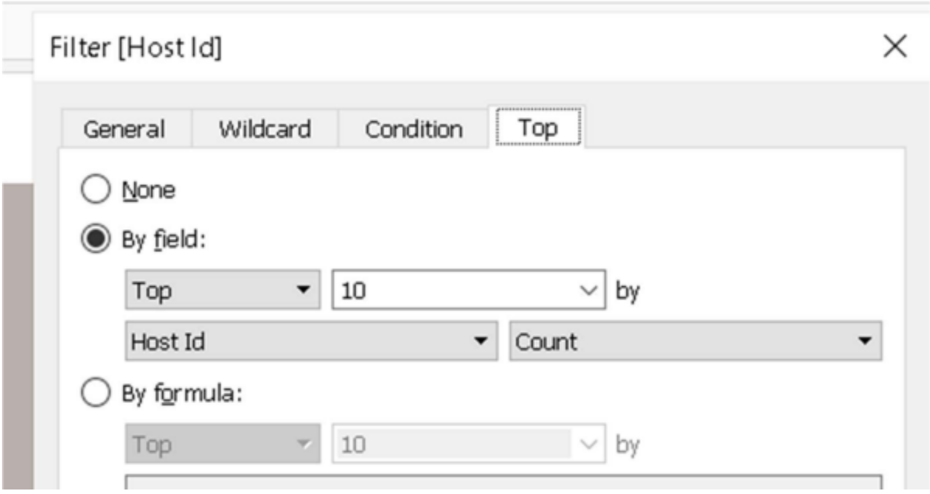
- Duplicate rows were checked in our dataset and no duplicate data was found. \
- Null Values were checked in our dataset. Columns like name, host-name, last review and review-per-month have null values.
- Dropped the column name as missing values are less and dropping it won't have significant impact on analysis.
- Formatting was checked in our dataset.
- Outliers were identified and reviewed.

Data Analysis and Visualizations using Tableau:

Tableau was used to visualize the data for the assignment. Below are the detailed steps used for each visualization.

1) **Top 10 Host:**

- The top 10 Host Ids were identified, 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

2) **Preferred Room type with respect to Neighbourhood group:**

- A Pie Chart was created for understanding the percentage of room type preferred with respect to 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) **Neighbourhood Groups Price variance:**

- A box and whisker's plot was used with Neighbourhood Groups in Columns and Price in Rows.

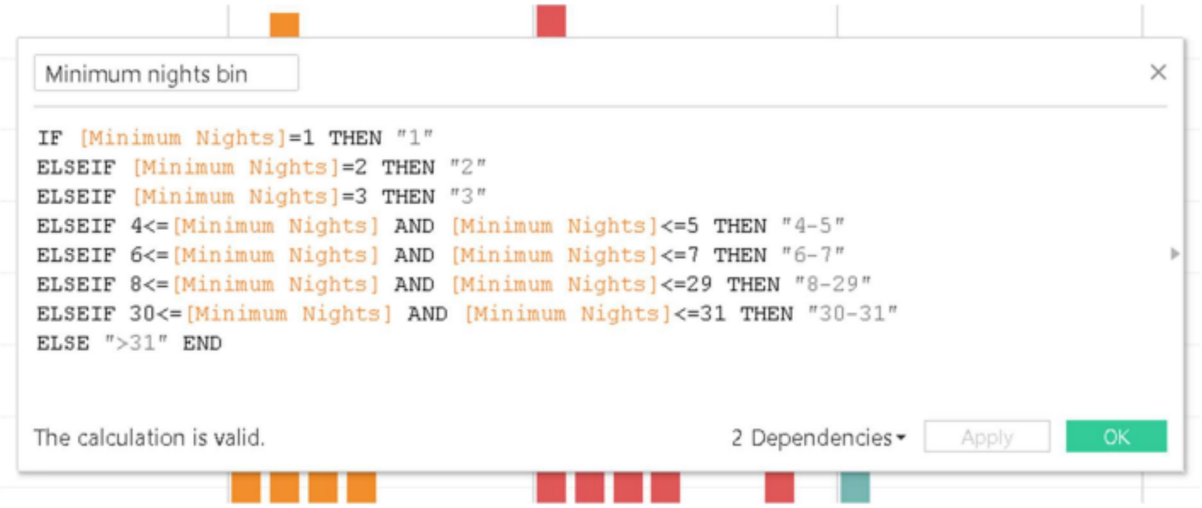
- The Price was changed from a Sum Measure to the median measure.

4) **Neighbourhood groups Average price :**

- A Bubble chart was created 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. We have also Put Avg price in Label.

5) 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.

6) 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.

7) Neighbourhood vs Availability:

- Dual axis chart was created using Bar Chart for availability of 365 and a line chart for price of top 10 neighbourhood group sorted by price.

Methodology Document PPT 2:

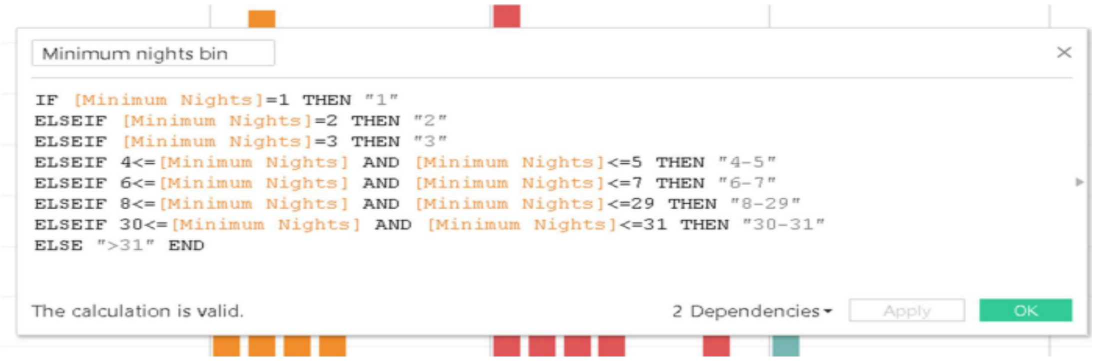
1) **Room type with respect to Neighbourhood group**:

   - A Pie Chart was created to understand the percentage of room type preferred with respect to the neighbourhood group

   - Room Type was added to the colours Marks card to highlight the different Room Type in different colours and count of Host Id to the size

2) **Customer Booking with respect to 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.

3) **Availability** vs **Neighbourhood :**

   - Dual axis chart was created using bar chart for availability 365 and line chart for price for top 10 neighbourhood group sorted by price.

4) **Price Range preferred by Customers:**

   - Pricing preference was taken 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.

5) Understanding Price variation with respect to Room Type & Neighbourhood:

- Highlights Table chat was created 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.

6) Price variation w r t Geography:

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

7) Popular Neighborhoods:

- We took neighbourhood in rows and sum of reviews in column and took neighbourhood groups in colour.
- Filter has been used to show Top 20 neighbours as per the sum of reviews.

8) Tools used:

- Data cleaning and preparation: Jupyter notebook – Python
- Visualization and analysis: Tableau
- Data Storytelling: Microsoft PPT