General Guidelines

↳ 1 cell hidden

Project Name - *Airbnb Bookings Analysis: Uncovering Hidden Insights in NYC*

Project Type - Exploratory Data Analysis (EDA)

Contribution - Individual

Name - Neha Gupta

Okay, before going to start. Let's understand what is Airbnb?

Looks Airbnb has interesting breakdown with names likely: Air, Bed and Breakfast to become Airbnb. Wow! This San-Francisco based startup offers you someone's home as a place to stay instead of a hotel. Looks, somewhat on a same business as OYO but the former doesn't owns any property instead acts as an intermediary between those who want to rent out space and those who are looking for space to rent.

Well, enough of it we understood what is the data all about and where it came from.

Project Summary

This project focuses on analyzing the Airbnb NYC 2019 dataset, which contains detailed information about approximately 49,000 Airbnb listings across New York City. The dataset includes 16 columns, each providing specific insights into various aspects of the listings, such as their location, pricing, availability, and host details.

Objectives:

- Location Analysis: Understand the distribution of Airbnb listings across different neighborhoods and broader areas within NYC, such as Manhattan, Brooklyn, Queens, Bronx, and Staten Island.
- Pricing Trends: Analyze the pricing strategies employed by hosts, examining how prices vary by room type, neighborhood, and other factors.
- Booking Requirements: Explore the minimum stay requirements set by hosts and how they influence guest booking patterns.
- Review Insights: Evaluate the number and frequency of reviews to gauge guest satisfaction and the popularity of listings.
- Host Activity: Investigate host behavior by analyzing the number of properties managed by each host and their overall activity on the platform.
- Availability Patterns: Study the availability of listings throughout the year to identify trends in booking availability and potential seasonal effects.

Expected Outcomes:

- A comprehensive understanding of how Airbnb operates in NYC, including key factors that influence pricing, booking behavior, and host performance.
- Data-driven insights that can help Airbnb improve its platform, support hosts in optimizing their listings, and enhance guest experiences.
- Visualization and reporting of key findings to provide actionable recommendations for various stakeholders, including Airbnb's marketing, finance, and operations teams.

This project aims to leverage the rich data provided by Airbnb to uncover valuable insights that can drive strategic decisions and improve the overall effectiveness of the platform in one of the world’s most dynamic cities.

GitHub Link -

https://github.com/neha2gupta7/Airbnb_NYC_EDA.git

Problem Statement

These problem statements cover a comprehensive range of exploratory questions and analyses that will provide valuable insights into the Airbnb NYC 2019 dataset. Here’s a brief outline of how you might approach each of these:

1. ***Distribution of Listings Across Neighborhoods***:

- Create a map or bar chart showing the number of listings per neighborhood.
- Compare listings across the five boroughs and within neighborhoods to find hotspots.

2. ***Price Variation by Neighborhood and Room Type***:

- Use box plots or violin plots to show price distributions by neighborhood and room type.
- Calculate average prices and identify neighborhoods with higher or lower prices.

3. ***Patterns in Minimum Night Stays***:

- Analyze the distribution of minimum night stays using histograms or bar charts.
- Investigate if certain neighborhoods or room types have longer minimum stays.

4. ***Availability Trends Throughout the Year***:

- Plot the availability_365 data over time to identify seasonal patterns.
- Check if there are specific months or seasons with higher or lower availability.

5. ***Highly Reviewed Listings by Neighborhood***:

- Create a ranking of neighborhoods based on the number of reviews and reviews_per_month.
- Identify neighborhoods with high guest feedback and explore potential reasons.

6. ***Host Activity and Listing Performance***:

- Analyze the relationship between calculated_host_listings_count and performance metrics like price, reviews, and availability.
- Determine if hosts with multiple listings have different performance compared to single-listing hosts.

7. ***Outliers in Pricing***:

- Use scatter plots or box plots to identify listings with unusual prices.
- Investigate factors contributing to these outliers, such as special amenities or locations.

8. ***Recent Reviews Across Neighborhoods***:

- Analyze trends in last_review data to see if recent reviews reflect changes in guest satisfaction.
- Compare recent reviews across different neighborhoods.

9. ***Factors Correlated with Higher Prices***:

- Perform a correlation analysis or build regression models to find variables most associated with higher prices.
- Look at the significance and strength of correlations.

10. ***Distribution of Room Types in Neighborhoods***:

- Examine the proportion of different room types per neighborhood using pie charts or stacked bar charts.
- Analyze how room type distribution affects pricing and availability. These analyses will help you uncover actionable insights about Airbnb listings in NYC and understand key factors influencing prices and guest satisfaction.

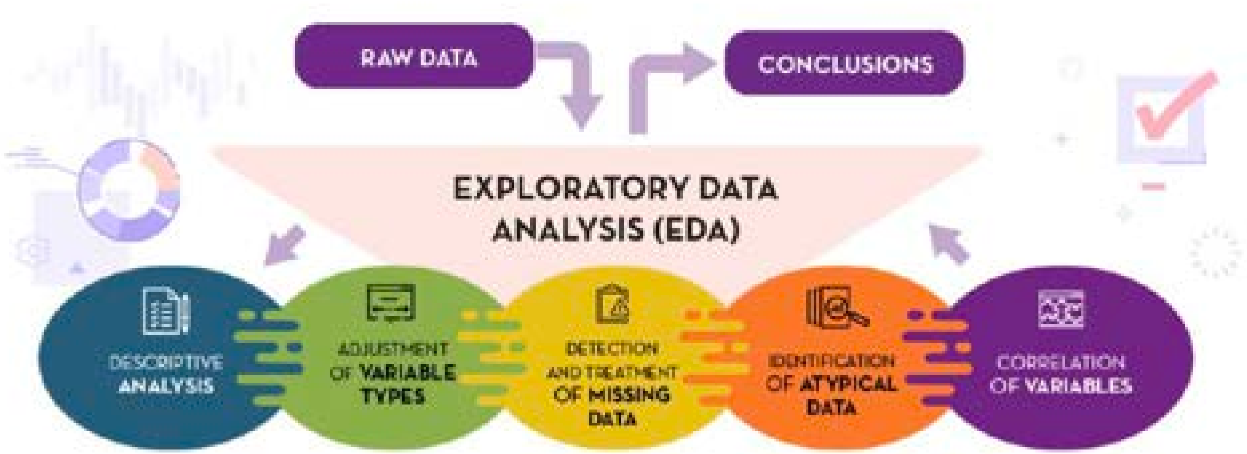
Define Your Business Objective?

To provide actionable insights for optimizing Airbnb listings in New York City by identifying key factors influencing pricing, availability, and guest satisfaction, thereby enhancing the performance of hosts and improving the overall Airbnb platform experience.

1. Optimize Pricing Strategies
2. Enhance Listing Visibility and Appeal
3. Improve Guest Satisfaction
4. Identify High-Performance Hosts and Listings
5. Detect and Address Outliers
6. Understand Neighborhood Dynamics
7. Support Strategic Decision-Making

By achieving these objectives, Airbnb can enhance platform functionality, improve host performance, and provide better experiences for guests, ultimately driving increased bookings and customer satisfaction.

Let's Begin !


RAW DATA
CONCLUSIONS
EXPLORATORY DATA ANALYSIS (EDA)
DESCRIPTIVE ANALYSIS
ADJUSTMENT OF VARIABLE TYPES
DETECTION AND TREATMENT OF MISSING DATA
IDENTIFICATION OF ATYPICAL DATA
CORRELATION OF VARIABLES

1. ***Know Your Data***

Import important Libraries

```
# Import Libraries
import numpy as np                  # Importing NumPy for numerical operations and array
import pandas as pd                 # Importing Pandas for data manipulation and analysis
from numpy import mean              # Importing the mean function from NumPy for calculat
import matplotlib.pyplot as plt     # Importing Matplotlib for creating static, animated,

# Ensures that plots are displayed inline in Jupyter notebooks
%matplotlib inline

import seaborn as sns               # Importing Seaborn for advanced data visualization,
import warnings
warnings.filterwarnings('ignore')   # Suppressing warnings to keep the output clean and f
```

Dataset Loading (AirBnB)

```
# Load Dataset
from google.colab import drive     # Importing the drive module from Google Colab to inte
drive.mount('/content/drive')      # Mounting Google Drive to the Colab environment to ac
```

```
Mounted at /content/drive
```

```
#File path of Airbnb dataset in google drive
file_path = "/content/drive/MyDrive/AlmaBetter Projects/Module 2/Capstone Project: Explor
Airbnb_df = pd.read_csv(file_path + "Airbnb NYC 2019.csv")
```

Dataset First View

```
# Dataset First Look
Airbnb_df
```

	id	name	host_id	host_name	neighbourhood_group	neighbou
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kens
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Mi
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	H
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clint
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East H
...	...	...	...	...	...	
48890	36484665	Charming one bedroom - newly renovated rowhouse	8232441	Sabrina	Brooklyn	Be Stuy
48891	36485057	Affordable room in Bushwick/East Williamsburg	6570630	Marisol	Brooklyn	Bus
48892	36485431	Sunny Studio at Historical Neighborhood	23492952	Ilgar & Aysel	Manhattan	H
48893	36485609	43rd St. Time Square-cozy single bed	30985759	Taz	Manhattan	Hell's K
48894	36487245	Trendy duplex in the very heart of Hell's Kitchen	68119814	Christophe	Manhattan	Hell's K

48895 rows × 16 columns

Next steps: Generate code with Airbnb_df View recommended plots New interactive sheet

Data Exploration and variable Identification:

Dataset Rows & Columns count

```
# Dataset Rows & Columns count
Airbnb_df.shape
```

```
(48895, 16)
```

dataset Airbnb_df has 48,895 rows and 16 columns

Dataset Information

```
# first 5 rows
Airbnb_df.head().T
```

	0	1	2	3	
id	2539	2595	3647	3831	5
name	Clean & quiet apt home by the park	Skylit Midtown Castle	THE VILLAGE OF HARLEM....NEW YORK !	Cozy Entire Floor of Brownstone	Entire Spac Studio/ by cei |
host_id	2787	2845	4632	4869	7
host_name	John	Jennifer	Elisabeth	LisaRoxanne	La
neighbourhood_group	Brooklyn	Manhattan	Manhattan	Brooklyn	Manha
neighbourhood	Kensington	Midtown	Harlem	Clinton Hill	I Hai
latitude	40.64749	40.75362	40.80902	40.68514	40.79
longitude	-73.97237	-73.98377	-73.9419	-73.95976	-73.94
room_type	Private room	Entire home/apt	Private room	Entire home/apt	Ei home
price	149	225	150	89	
minimum_nights	1	1	3	1	
number_of_reviews	9	45	0	270	
last_review	2018-10-	2019-05-	NaN	2019-07-05	2018

Next steps: Generate code with Airbnb_df | View recommended plots | New interactive sheet

this display the first 5 rows of dataset, but transposed, so that the rows become columns and vice versa. Each column name will be shown in the first row, with the corresponding data values below it.

```
# last 5 rows
Airbnb_df.tail()
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhoo
48890	36484665	Charming one bedroom - newly renovated rowhouse	8232441	Sabrina	Brooklyn	Bedfor Stuyvesa
48891	36485057	Affordable room in Bushwick/East Williamsburg	6570630	Marisol	Brooklyn	Bushwic
48892	36485431	Sunny Studio at Historical Neighborhood	23492952	Ilgar & Aysel	Manhattan	Harle
48893	36485609	43rd St. Time Square-cozy single bed	30985759	Taz	Manhattan	Hell's Kitche
48894	36487245	Trendy duplex in the very heart of Hell's Kitchen	68119814	Christophe	Manhattan	Hell's Kitche

```
#basic information about the dataset
Airbnb_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
```

```
#check data types
Airbnb_df.dtypes
```

```
                                          0
                          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
```

So, host_name, neighbourhood_group, neighbourhood and room_type fall into categorical variable category.

While host_id, latitude, longitude, price, minimum_nights, number_of_reviews, last_review, reviews_per_month, host_listings_count, availability_365 are numerical variables

Duplicate Values

```
# Dataset Duplicate Value Count
Airbnb_df.duplicated().sum()              # This is useful for identifying how many d
Airbnb_df_dup = Airbnb_df.drop_duplicates()   # This ensures that Airbnb_df_dup contains
Airbnb_df_dup.count()                     # This verifies the number of non-null entr
```

	0
id	48895
name	48879
host_id	48895
host_name	48874
neighbourhood_group	48895
neighbourhood	48895
latitude	48895
longitude	48895
room_type	48895
price	48895
minimum_nights	48895
number_of_reviews	48895
last_review	38843
reviews_per_month	38843
calculated_host_listings_count	48895
availability_365	48895

dtype: int64

Great! We've confirmed that dataset has no duplicate rows after checking and removing duplicates. This ensures the integrity of data for analysis.

Rename columns:

```
rename_col = {'id':'listing_id','name':'listing_name','number_of_reviews':'total_reviews'


# use a pandas function to rename the current function
Airbnb_df = Airbnb_df.rename(columns = rename_col)
Airbnb_df.head(2)
```

	listing_id	listing_name	host_id	host_name	neighbourhood_group	neighbourhood
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown

Next steps: Generate code with Airbnb_df View recommended plots New interactive sheet

Missing Values/Null Values

```
# Missing Values/Null Values Count
Airbnb_df.isnull().sum()
```

	0
listing_id	0
listing_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
total_reviews	0
last_review	10052
reviews_per_month	10052
host_listings_count	0
availability_365	0

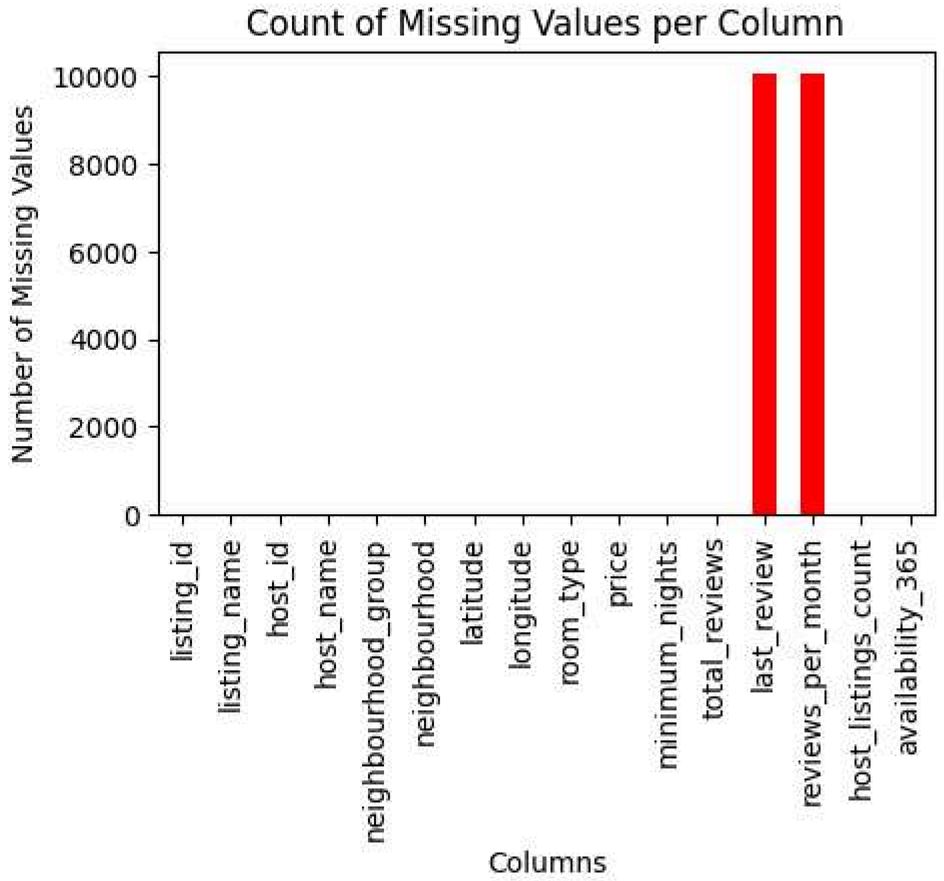
dtype: int64

In dataset, the null values are distributed across various columns, as indicated by the count of missing entries for each column:

- last_review: Contains 10,052 missing values.
- reviews_per_month: Also contains 10,052 missing values.
- name: have 16 missing values
- host_name: have 21 missing values

The other columns, such as id, neighbourhood_group, neighbourhood, latitude, longitude, room_type, price, minimum_nights, number_of_reviews, calculated_host_listings_count, and availability_365, do not have any missing values.

```
# Step 1: Calculate missing values for each column
# This line creates a Series where the index is the column names,
# and the values are the count of missing (NaN) values in each column.
missing_values = Airbnb_df.isnull().sum()

# Step 2: Set up the figure size for the plot
# 'plt.figure()' creates a new figure object for the plot.
# 'figsize=(5, 3)' defines the width and height of the figure in inches.
plt.figure(figsize=(5, 3))

# Step 3: Plot the missing values as a bar chart
# 'missing_values.plot(kind="bar")' creates a bar chart where the x-axis
# represents the columns and the y-axis represents the count of missing values.
# 'color="red"' specifies that the bars should be red to highlight missing values.
missing_values.plot(kind='bar', color='red')

# Step 4: Add a title to the plot
# 'plt.title()' sets the title of the plot. This helps to clarify the purpose
# of the chart for anyone viewing it.
plt.title('Count of Missing Values per Column')

# Step 5: Label the axes
# 'plt.xlabel()' and 'plt.ylabel()' label the x and y axes respectively,
# so viewers can easily understand what each axis represents.
plt.xlabel('Columns')
plt.ylabel('Number of Missing Values')

# Step 6: Rotate the x-axis labels to avoid overlapping
# 'plt.xticks(rotation=90)' rotates the labels on the x-axis by 90 degrees
# to ensure they don't overlap and remain readable.
plt.xticks(rotation=90)

# Step 7: Display the plot
# 'plt.show()' renders the plot to the output, making it visible.
plt.show()
```



The chart displays the count of missing values per column, with the x-axis representing the columns and the y-axis representing the number of missing values.

What did you know about your dataset?

Here's a breakdown of what each column in the Airbnb NYC 2019 dataset represents:

1. Columns and Missing Values:

- **Missing values:**
  - last_review and reviews_per_month have 10,052 missing values each.
  - Other columns do not have any missing values.

2. Columns Overview:

- **listing_id**: A unique identifier for each listing. Every Airbnb listing has a distinct ID.
- **listing_name**: The name or title of the listing, which is often chosen by the host to attract guests.
- **host_id**: A unique identifier for each host. If a host has multiple listings, they will all share the same host ID.

- **host_name**: The name of the host who owns or manages the listing.
- **neighbourhood_group**: The broader area or district in NYC where the listing is located, such as Manhattan, Brooklyn, Queens, Bronx, or Staten Island.
- **neighbourhood**: The specific neighborhood within the broader group where the listing is situated, providing more granular location information.
- **latitude**: The geographic latitude coordinate of the listing, useful for mapping and spatial analysis.
- **longitude**: The geographic longitude coordinate of the listing, also useful for mapping and spatial analysis.
- **room_type**: The type of room being offered in the listing, such as an entire home/apartment, a private room, or a shared room.
- **price**: The nightly price in USD that guests must pay to stay at the listing.
- **minimum_nights**: The minimum number of nights a guest is required to book to stay at the listing.
- **total_reviews**: The total number of reviews the listing has received from guests.
- **last_review**: The date of the most recent review left by a guest for the listing.
- **reviews_per_month**: The average number of reviews the listing receives per month.
- **host_listings_count**: The total number of listings that a host has on Airbnb. This helps identify whether the host is managing multiple properties.
- **availability_365**: The number of days within a year that the listing is available for booking. This ranges from 0 (not available) to 365 (available every day of the year).

3. Dataset Size:

- Contains around 49,000 observations and 16 columns.

4. Data Types:

- The dataset includes a mix of numerical (e.g., price, latitude) and categorical (e.g., room_type, neighbourhood_group) data.

These details provide a foundation for exploring and analyzing the dataset to gain insights into Airbnb listings in New York City.

host_name and **listing_name** are not that much of null values, so first we are good to fill those with some substitutes in both the columns first.

```
Airbnb_df['listing_name'].fillna('unknown',inplace=True)
Airbnb_df['host_name'].fillna('no_name',inplace=True)
```

```
#so the null values are removed
Airbnb_df[['host_name','listing_name']].isnull().sum()
```

```
                 0
host_name        0
listing_name     0

dtype: int64
```

To handle the missing values in the listing_name and host_name columns, we've replaced them with 'unknown' and 'no_name', respectively.

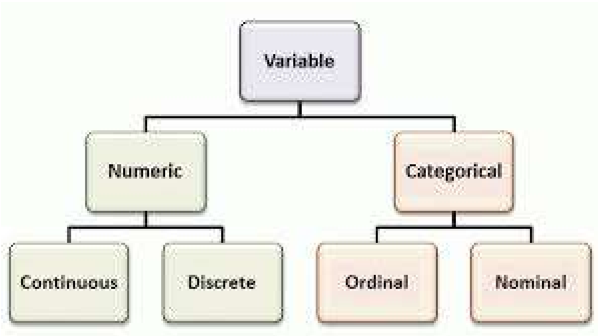
now, the columns **last_review** and **reviews_per_month** have total 10052 null values each.

```
# Fill missing values for 'reviews_per_month' and 'last_review'
# Since there are 10,052 missing values for 'reviews_per_month', we assume listings witho
Airbnb_df['reviews_per_month'].fillna(0, inplace=True)

# For 'last_review', we fill missing values with a placeholder date
# taken date 1970-01-01 as default value which is not on Airbnb Data
Airbnb_df['last_review'].fillna('1970-01-01', inplace=True)
```

```
#check again for missing values
Airbnb_df[['last_review','reviews_per_month']].isnull().sum()
```

```
                      0
last_review           0
reviews_per_month     0

dtype: int64
```

2. Understanding Your Variables



Our first job is to explore the variables that are present in our dataset. We have discussed every relevent column in detail and have also explored unique data present in each relevant column of our dataset.

```
#checking what are the variables here:
#Dataset Columns
Airbnb_df.columns
```

```
Index(['listing_id', 'listing_name', 'host_id', 'host_name',
       'neighbourhood_group', 'neighbourhood', 'latitude', 'longitude',
       'room_type', 'price', 'minimum_nights', 'total_reviews', 'last_review',
       'reviews_per_month', 'host_listings_count', 'availability_365'],
      dtype='object')
```

```
# Numerical columns(int64,float64)
num_data = Airbnb_df.select_dtypes(include=['int64','float64']).columns
num_data
```

```
Index(['listing_id', 'host_id', 'latitude', 'longitude', 'price',
       'minimum_nights', 'total_reviews', 'reviews_per_month',
       'host_listings_count', 'availability_365'],
      dtype='object')
```

```
# categorical columns(object)
cat_data = Airbnb_df.select_dtypes(include=['object']).columns
cat_data
```

```
Index(['listing_name', 'host_name', 'neighbourhood_group', 'neighbourhood',
       'room_type', 'last_review'],
      dtype='object')
```

```
# Generate descriptive statistics for non-numerical (categorical) columns
Airbnb_df.describe(include='object')
```

	listing_name	host_name	neighbourhood_group	neighbourhood	room_type	last_
count	48895	48895	48895	48895	48895	
unique	47906	11453	5	221	3	
top	Hillside Hotel	Michael	Manhattan	Williamsburg	Entire home/apt	197

Key Observations:

- High Number of Unique Values:

listing_name and host_name have a large number of unique values, which indicates high variability and might be less useful for categorical analysis unless aggregated.

- Frequent Categories:

neighbourhood_group and room_type have fewer unique values and a dominant category, which can simplify analysis.

- Frequent Last Review Date:

A specific date appears frequently in the last_review column, indicating that many reviews might be clustered around certain times.

Handling Categorical Data:

- Encoding: Convert categorical variables into numerical format using techniques like one-hot encoding or label encoding.
- Aggregation: For high cardinality features (e.g., listing_name, host_name), consider aggregating or binning categories to reduce dimensionality.

If you need further analysis or have specific questions about these categorical features, let me know!

```
# Dataset Describe
# Summary statistical analysis of numerical columns
Airbnb_df.describe()
```

	listing_id	host_id	latitude	longitude	price	minimum_ni
count	4.889500e+04	4.889500e+04	48895.000000	48895.000000	48895.000000	48895.00
mean	1.901714e+07	6.762001e+07	40.728949	-73.952170	152.720687	7.02
std	1.098311e+07	7.861097e+07	0.054530	0.046157	240.154170	20.51
min	2.539000e+03	2.438000e+03	40.499790	-74.244420	0.000000	1.00
25%	9.471945e+06	7.822033e+06	40.690100	-73.983070	69.000000	1.00
50%	1.967728e+07	3.079382e+07	40.723070	-73.955680	106.000000	3.00
75%	2.915218e+07	1.074344e+08	40.763115	-73.936275	175.000000	5.00
max	3.648724e+07	2.743213e+08	40.913060	-73.712990	10000.000000	1250.00

To get a statistical summary of the dataset, including measures like count, mean, standard deviation, min, max, and percentiles, you can use the describe() method in pandas. This method will provide insights into the distribution and range of your numerical columns.

Variables Description

This will output a table with the following information for each numerical column:

- count: The number of non-null entries.
- mean:The average value.

- std: The standard deviation, showing the spread of the data.
- min: The minimum value.
- 25%: The 25th percentile (first quartile).
- 50%: The 50th percentile (median or second quartile).
- 75%: The 75th percentile (third quartile).
- max: The maximum value.

This summary is useful for understanding the overall structure and distribution of data, identifying outliers, and getting a sense of the range and central tendencies of all variables.

Key Observations:

- Price: The price varies significantly with a mean around dollar 152 but a high standard deviation indicating wide variability. The maximum price is quite high at $10,000.
- Minimum Nights: While most listings require just a few nights (median of 3), some have extremely high minimum night requirements (up to 1,250).
- Reviews per Month: There's a significant variation, with most listings receiving fewer than 2 reviews per month.
- Availability: A substantial number of listings are either fully booked or rarely available, as indicated by the median of 45 days of availability per year.

Check Unique Values for each variable.

```
# check unique values for listing/property Ids
# all the listing ids are different and each listings are different here.
Airbnb_df['listing_id'].nunique()
```

```
48895
```

```
# so there are 221 unique neighborhood in Dataset
Airbnb_df['neighbourhood'].nunique()
```

```
221
```

```
#and total 5 unique neighborhood_group in Dataset
Airbnb_df['neighbourhood_group'].nunique()
```

```
5
```

```
#so total 11453 different hosts in Airbnb-NYC
Airbnb_df['host_name'].nunique()
```

```
11453
```

```
# most of the listing/property are different in Dataset
Airbnb_df['listing_name'].nunique()
```

```
47906
```

Note - so i think few listings/property with same names has different hosts in different areas/neighbourhoods of a neighbourhood_group

```
#find distinct in all categorical columns
for x in cat_data:
  print(x)
  print("\n")
  print(Airbnb_df[x].unique())
  print("\n")
  print("---------------------------------------")
  print(Airbnb_df[x].value_counts())
  print("\n")
  print('********************************************************************************
[ '2018-10-19' '2019-05-21' '1970-01-01' ... '2017-12-25' '2018-01-29'
 '2018-03-29']


---------------------------------------
last_review
1970-01-01    10052
2019-06-23     1413
2019-07-01     1359
2019-06-30     1341
2019-06-24      875
              ...
2015-01-12        1
2014-09-17        1
2015-02-12        1
2014-07-10        1
2014-08-15        1
Name: count, Length: 1765, dtype: int64


*********************************************************************************
```

```
#find distinct in all numerical columns
for y in num_data:
  print(y)
  print("\n")
  print(Airbnb_df[y].unique())
  print("\n")
  print("---------------------------------------")
  print(Airbnb_df[y].value_counts())
  print("\n")
  print('*********************************************************************************
      341 244 329 255 348   2  56  68 360  76  15 226 349  11 316 281 287  14
       86 261 331  51 254 103  42 325  35 203   5 276 102  71  78   8 182  79
       49 156 200 106 135  81 142 179  52 237 204 181 296 335 282 274  98 157
      174 223 361 283 315  36 271 139 193 136 277 221 264 236  89  23 218 235
      119 350 161 259  27 167 358  59 337  43  25 127 303 115 268  44  65 252
       64 111  90 338  31 241 285 183  84 166  28  83 305 356 308 229 210 153
      332 120 313  69 293   4 300  40 117 206 144 354  41 270 306  33  50  80
       97 118 134  17 289 121 205  74  62  29 109 168 146 242 352 155 291 266
      101 190 327 217 171 110  87 202  70 147 169 212 122 330  54 196  57  73
      149 239  63 195  47 319  19 112 344  77 160 141  13  24 150 128 176 357
      211 172 256 165  32 105 267 148  93  45 175 159  48 100 184 114 133 186
      334  94 151 228 113  55  66 173 104 197  99 131 143 124 130 187 145 108
      123  92  61 138 227  82]


     ---------------------------------------
     availability_365
     0      17533
     365     1295
     364      491
     1        408
     89       361
            ...
     195       26
     183       24
     196       24
     181       23
     202       20
     Name: count, Length: 366, dtype: int64


     **********************************************************************************
```

```
#looks there are few listings where the property name and the host have same names!
Airbnb_df[Airbnb_df['listing_name'] == Airbnb_df['host_name']].head()
```

	listing_id	listing_name	host_id	host_name	neighbourhood_group	neighbour
9473	7264659	Olivier	6994503	Olivier	Manhattan	Upper
10682	8212051	Monty	43302952	Monty	Brooklyn	East Fla
16422	13186374	Sean	35143476	Sean	Brooklyn	Wi Te
23996	19348168	Cyn	74033595	Cyn	Brooklyn	Bec Stuyv
24152	19456810	Hillside Hotel	134184451	Hillside Hotel	Queens	Briar

```
# Same host have hosted different listing/property in different or same neighbourhood in
Airbnb_df.loc[(Airbnb_df['neighbourhood_group']=='Queens') & (Airbnb_df['host_name']=='Al
```

	listing_id	listing_name	host_id	host_name	neighbourhood_group	neighbourh
3523	2104910	SPACIOUS APT BK/QUEENS w/BACKYARD!	10643810	Alex	Queens	Ridgev
4512	3116519	Large 900 sqft Artist's Apartment	3008690	Alex	Queens	Ridgev
6178	4518242	Zen MiniPalace Astoria	23424461	Alex	Queens	As
10543	8090529	Modern studio in Queens, NY	17377835	Alex	Queens	Sunny
10681	8211513	Cozy Room, In Quiet Neighborhood	43299973	Alex	Queens	Ridgev

So, far I was trying to understand more deep on the two variables: listing_name and host_name & its relationship with neighbourhood_group and neighbourhood.(only from the values present inside)

Found out that: A host can have multiple properties in a neighbourhood group with different host-ids but a host with a particular property/listing in a particular neighbourhood of a neighbourhood group have a same host-id(not mandatory as there are exceptions where few hosts have diferrent id's for each listing/property in a neighbourhood)

Also the data so far tells, there might be cases where a particular host has co-hosted someone else's property/listing in a neighbourhood on Airbnb.

We'll not bother much as these are not that important in our analysis and proceed further!

3. ***Data Wrangling***

Discovering
Structuring
Cleaning
Enriching
Validating
Publishing
Data Wrangling

Data Wrangling Code

1. Extract the date, month, and year from the last_review column

```
# Step 1: Convert the last_review column is in datetime format
Airbnb_df['last_review'] = pd.to_datetime(Airbnb_df['last_review'])
Airbnb_df['last_review'].head()
```

```
     last_review
0    2018-10-19
1    2019-05-21
2    1970-01-01
3    2019-07-05
4    2018-11-19

dtype: datetime64[ns]
```

```
# Step 2: Extract day, month, and year
Airbnb_df['review_day'] = Airbnb_df['last_review'].dt.day
Airbnb_df['review_month'] = Airbnb_df['last_review'].dt.month
Airbnb_df['review_year'] = Airbnb_df['last_review'].dt.year


# Step 3: review all four columns
Airbnb_df.loc[ :,['last_review','review_day','review_month','review_year']].head()
```

	last_review	review_day	review_month	review_year
0	2018-10-19	19	10	2018
1	2019-05-21	21	5	2019
2	1970-01-01	1	1	1970
3	2019-07-05	5	7	2019
4	2018-11-19	19	11	2018

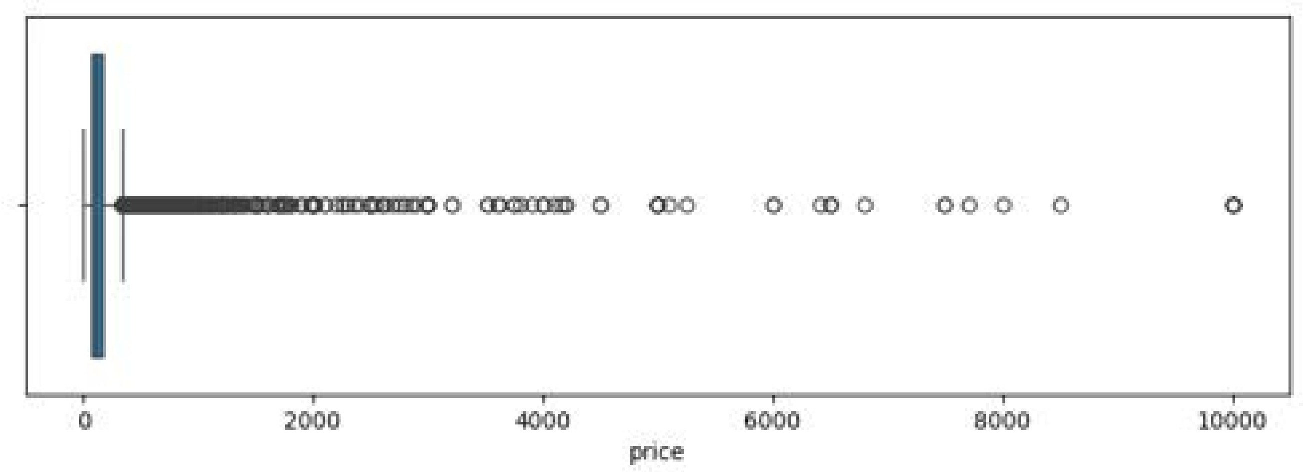
2. Removing outliers

- **Note** - price column is very important so we have to find big outliers in important columns first.

```
# so 11 property/listings have 0 price listed
len(Airbnb_df[Airbnb_df['price']==0])
```

```
11
```

```
# use box plot for price
plt.figure(figsize=(10,3))    # This line of code sets up the figure
sns.boxplot(x = Airbnb_df['price'])   # Seaborn function, This creates the actual box plo
plt.show()   # This line of code displays the figure and plot that were created by the pr
```

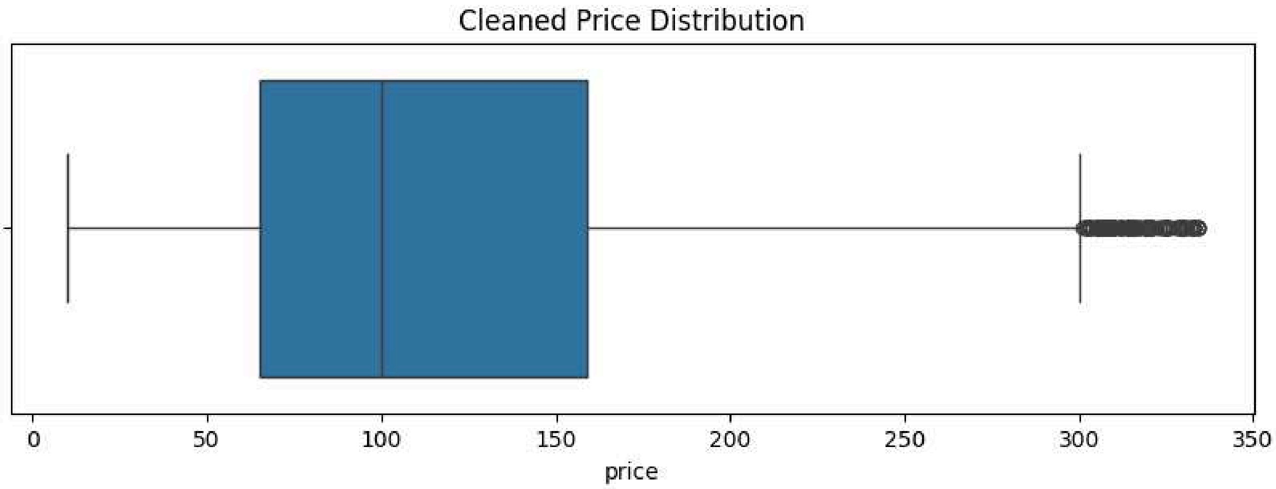


- There are significant outliers at the higher end of the price distribution.

- Extremely high prices are pulling the distribution to the right (right skew).
- There's a dense cluster of prices between dollar 0 and dollar 500.
- From the boxplot, it seems like there are some prices near $0, which we have already noted as needing to be removed from the dataset.

- **Using IQR technique**

```
# Step 1: Remove properties with zero price
Airbnb_df_cleaned = Airbnb_df[Airbnb_df['price'] > 0]


# Step 1: Calculate the first quartile (Q1) for the 'price' column
# 'quantile(0.25)' returns the 25th percentile (Q1), which is the value below which 25% o
Q1 = Airbnb_df_cleaned['price'].quantile(0.25)

# Step 2: Calculate the third quartile (Q3) for the 'price' column
# 'quantile(0.75)' returns the 75th percentile (Q3), which is the value below which 75% o
Q3 = Airbnb_df_cleaned['price'].quantile(0.75)

# Step 3: Calculate the interquartile range (IQR)
# The IQR is the difference between Q3 and Q1, representing the range of the middle 50% o
# It is a measure of statistical dispersion, showing where most of the data is concentrat
IQR = Q3 - Q1

# Step 4: Define the lower bound for detecting outliers
# Outliers below this threshold are considered unusually low.
# The lower bound is calculated as Q1 minus 1.5 times the IQR.
# This rule of thumb (1.5 * IQR) is commonly used to detect mild outliers.
lower_bound = Q1 - 1.5 * IQR

# Step 5: Define the upper bound for detecting outliers
# Outliers above this threshold are considered unusually high.
# The upper bound is calculated as Q3 plus 1.5 times the IQR.
# Data points outside of this range are considered outliers.
upper_bound = Q3 + 1.5 * IQR


# Step 3: Remove properties with prices outside the IQR range (outliers)
Airbnb_df_cleaned = Airbnb_df_cleaned[(Airbnb_df_cleaned['price'] >= lower_bound) & (Airb



# Step 4: Visualize the cleaned price distribution
plt.figure(figsize=(10, 3))
sns.boxplot(x=Airbnb_df_cleaned['price'])
plt.title('Cleaned Price Distribution')
plt.show()
```



```
# Save the cleaned dataset
Airbnb_df_cleaned.to_csv(file_path + 'Airbnb_cleaned.csv', index=False)
```

What all manipulations have you done and insights you found?

1. **Convert last_review to DateTime and Extract Date, Month, and Year**:

- We can now analyze the seasonality of reviews by looking at trends across specific months or years.
- We can identify if there is a drop-off in reviews during certain years might reflect changes in Airbnb policies or external factors like COVID-19.

2. **Removing Outliers from the price Column**:

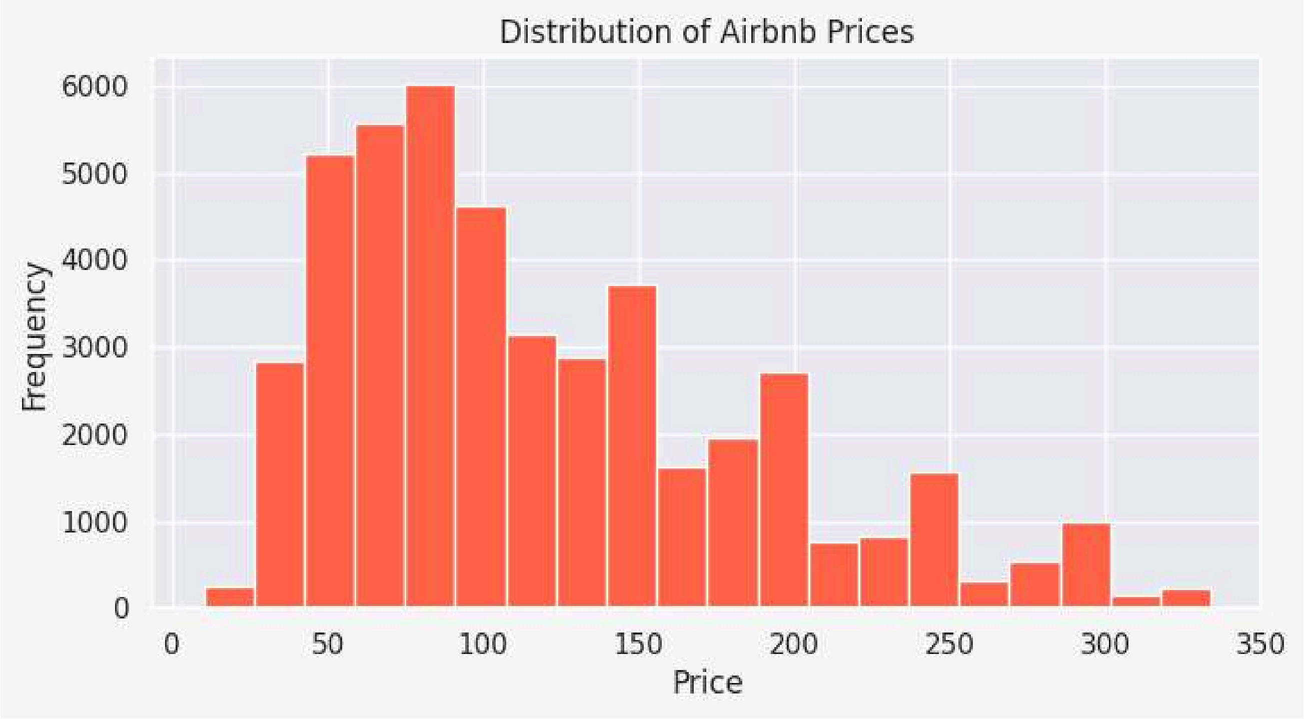
- Removing outliers helps identify a realistic price range for Airbnb listings in NYC.
- Outliers in the price column can skew analysis, so we will remove them by determining a reasonable threshold. A typical approach is to use the interquartile range (IQR) to identify outliers.
- Listings with extremely low or high prices (outside the IQR range) have been removed.
- We can now more accurately analyze how other features (e.g., room type, location) affect prices without being biased by outliers.
- Extreme outliers could represent luxury properties or incorrectly entered prices, and their removal ensures the analysis is focused on the general market.

4. Data Vizualization, Storytelling & Experimenting with charts : Understand the relationships between variables

Univariate Analysis (1 variable)

⌄ *Chart - 1* : Histogram - Shows the distribution of listing prices.

```
# Step 1: Set the Seaborn theme to 'darkgrid'
# 'darkgrid' adds a dark background with gridlines, making the plot easier to read.
sns.set_theme(style='darkgrid')

# Step 2: Adjust the figure size and background color
# 'plt.figure(figsize=(8, 4))' defines the size of the plot as 8 inches wide and 4 inches
# 'facecolor' sets the background color of the figure to a light greyish tone for better
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')

# Step 3: Create a histogram of the 'price' column
# 'bins=20' splits the price data into 20 intervals (bins).
# 'color=#FF6347' sets the color of the bars to a tomato-red shade using a hex color code
plt.hist(Airbnb_df_cleaned['price'], bins=20, color='#FF6347')

# Step 5: Add labels for the x and y axes
# 'plt.xlabel()' adds a label to the x-axis, which represents the price in this case.
plt.xlabel('Price')
# 'plt.ylabel()' adds a label to the y-axis, representing the frequency of occurrences in
plt.ylabel('Frequency')

# Step 6: Add a title to the histogram
# 'plt.title()' sets a descriptive title for the plot, giving context to what is being di
plt.title('Distribution of Airbnb Prices')

# Step 7: Display the plot
# 'plt.show()' renders the plot, making it visible in the output.
plt.show()
```



1. Why did you pick the specific chart?

I picked a histogram for the chart because it is an effective way to visualize the distribution of a continuous variable, such as listing prices. A histogram allows us to see the shape of the distribution, including the central tendency, dispersion, and any skewness or outliers. This helps to identify patterns and trends in the data that might not be immediately apparent from a table or summary statistics.

2. What is/are the insight(s) found from the chart?

The insight gained from the chart is that the distribution of listing prices is right-skewed, meaning that there are more listings at lower price points than at higher price points. This suggests that the majority of listings are concentrated in the lower price ranges, with fewer listings at higher price points.

3. Will the gained insights help creating a positive business impact?

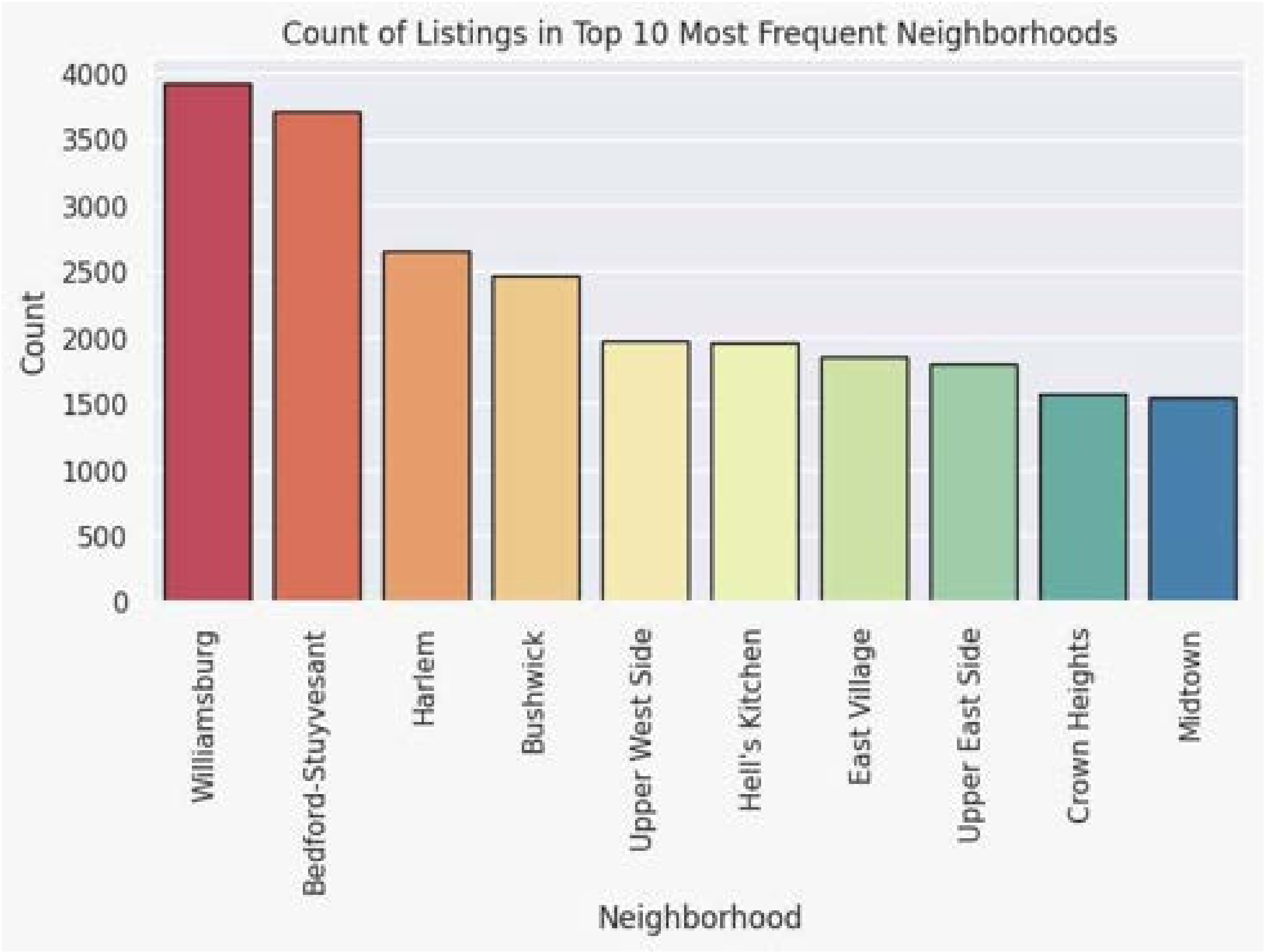
Are there any insights that lead to negative growth? Justify with specific reason.

- By understanding the distribution of listing prices, Airbnb can optimize its pricing strategy to maximize revenue. For example, they could focus on promoting listings in the most popular price ranges to attract more customers.

- The insights can also help Airbnb to identify opportunities to increase revenue by targeting specific price segments. For example, they could offer premium services or features to listings in higher price ranges to increase average revenue per user.
- Additionally, the insights can help Airbnb to improve its user experience by providing more relevant search results and recommendations. By understanding the distribution of listing prices, they can ensure that users are shown a diverse range of listings that meet their budget and preferences.
- One potential insight that could lead to negative growth is the concentration of listings in lower price ranges. This could lead to increased competition among hosts, driving prices down and reducing revenue for Airbnb. Additionally, the lack of listings in higher price ranges could limit Airbnb's ability to attract high-end travelers and reduce its average revenue per user. To mitigate these risks, Airbnb could consider strategies to incentivize hosts to list their properties at higher price points, such as offering premium services or features, or providing targeted marketing support.

Chart - 2 : Count Plot : Top 10 Most Frequent Neighborhoods

```
# Step 1: Get the top 10 most frequent neighborhoods
top_neighborhoods = Airbnb_df['neighbourhood'].value_counts().head(10).index

# Step 2: Filter the DataFrame to include only these top 10 neighborhoods
filtered_df = Airbnb_df[Airbnb_df['neighbourhood'].isin(top_neighborhoods)]

# Step 3: Create a Count Plot
plt.figure(figsize = (8, 4), facecolor='#F7F7F7')
sns.countplot(x='neighbourhood', data=filtered_df, order= top_neighborhoods, palette= 'Sp

# Set the title and labels
plt.title('Count of Listings in Top 10 Most Frequent Neighborhoods')
plt.xlabel('Neighborhood')
plt.ylabel('Count')

# Rotate x-axis labels for better readability
plt.xticks(rotation= 90)

# Display the plot
plt.show()
```



1. Why did you pick the specific chart?

I picked the specific chart (vertical bar chart) because it effectively compares quantities across different categories (neighborhoods in this case). It allows for easy visual comparison of listing counts, making it an ideal choice for displaying the count of listings in the top 10 most frequent neighborhoods.

2. What is/are the insight(s) found from the chart?

The key insights from the chart are:

- Williamsburg has the highest number of listings, significantly more than any other neighborhood.
- There's a gradual decrease in listing counts from the most frequent to least frequent neighborhoods.
- The top 3 neighborhoods (Williamsburg, Bedford-Stuyvesant and Harlem) have noticeably more listings than the others.
- East Harlem has the lowest number of listings among the top 10, but still close to 1000.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

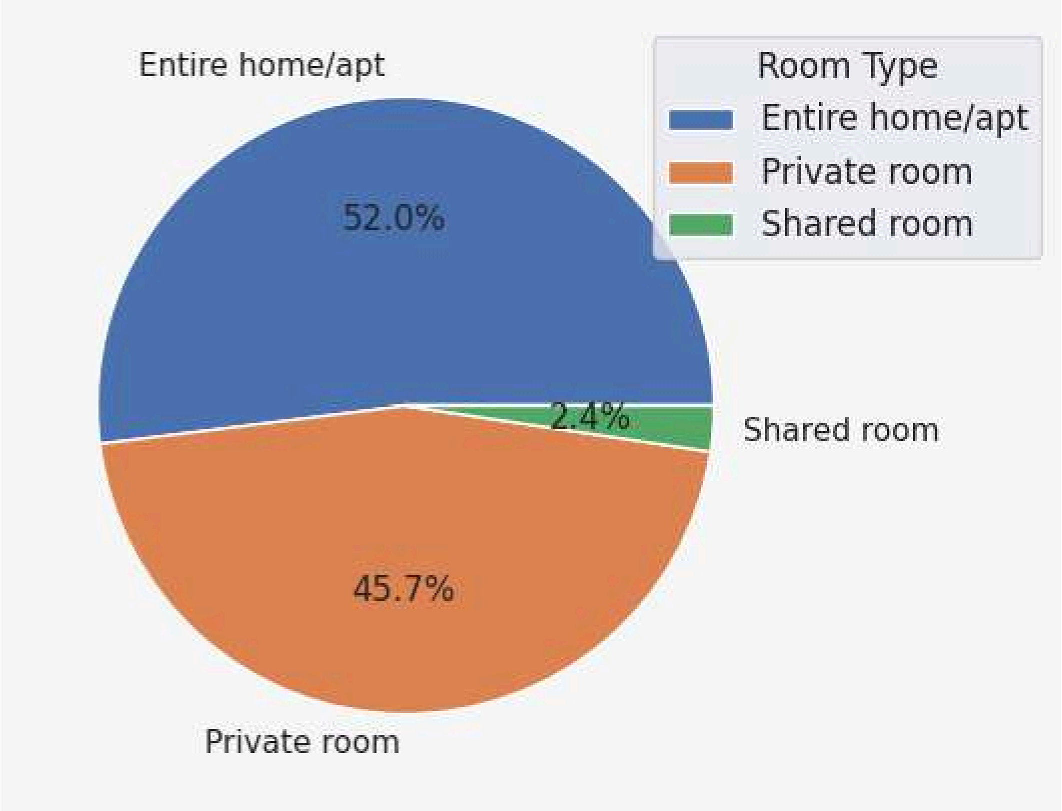
These insights can potentially create positive business impact:

- They help identify the most popular areas for listings, which could guide marketing efforts or investment decisions.
- Understanding the distribution of listings across neighborhoods can inform pricing strategies and help in resource allocation for property management.
- It may indicate areas of high demand, which could be useful for both hosts and the platform in expanding their services.

There are no direct insights that lead to negative growth.

Chart - 3 : Pie Chart : Distribution of Airbnb Listings by Room Type in NYC

```
# create a new DataFrame that displays the number of listings of each room type in the Ai
top_room_type = Airbnb_df['room_type'].value_counts().reset_index()

# rename the columns of the resulting DataFrame to 'Room_Type' and 'Total_counts'
top_room_type.columns = ['Room_Type', 'Total_counts']

# Set the figure size
plt.figure(figsize=(5, 5), facecolor='#F7F7F7')

# Get the room type counts
room_type_counts = Airbnb_df['room_type'].value_counts()

# Set the labels and sizes for the pie chart
labels = room_type_counts.index
sizes = room_type_counts.values

# Create the pie chart
plt.pie(sizes, labels=labels, autopct='%1.1f%%')

# Add a legend to the chart
plt.legend(title='Room Type', bbox_to_anchor=(0.8, 0, 0.5, 1), fontsize='12')

# Show the plot
plt.show()
```



1. Why did you pick the specific chart?

I chose a pie chart because:

- Categorical Data Representation: A pie chart is effective at showing the proportional breakdown of categorical data—in this case, different room types. It easily conveys the relative sizes of each category at a glance.
- Percentage View: Pie charts are excellent for displaying data in percentages, which helps in quickly understanding the proportion of listings of each room type in the dataset.

2. What is/are the insight(s) found from the chart?

- Dominance of Entire Homes/Apartments: More than half of the listings (52%) are for entire homes or apartments, indicating that many hosts prefer renting out entire spaces rather than just rooms.
- Significant Proportion of Private Rooms: A considerable portion (45.7%) of the listings are private rooms, showing that there's still a strong market for travelers looking for less expensive options or shared accommodations.
- Shared Rooms Are Rare: Only 2.4% of listings offer shared rooms, indicating that this option is far less popular in New York City's Airbnb market.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, these insights can be leveraged for business decisions:

- Targeted Marketing: Since entire homes/apartments make up the majority of listings, Airbnb can focus more of its marketing efforts on promoting these types of accommodations, especially to families and groups of travelers.
- Promoting Shared Rooms: Shared rooms have a very small market share. Airbnb could potentially create campaigns to promote this option as a budget-friendly alternative for solo travelers or those looking for a more social experience.
- Private Room Strategy: The almost equal split between private rooms and entire homes suggests a dual marketing approach—targeting both budget-conscious travelers looking for private rooms and those seeking a more luxurious experience in entire homes/apartments.

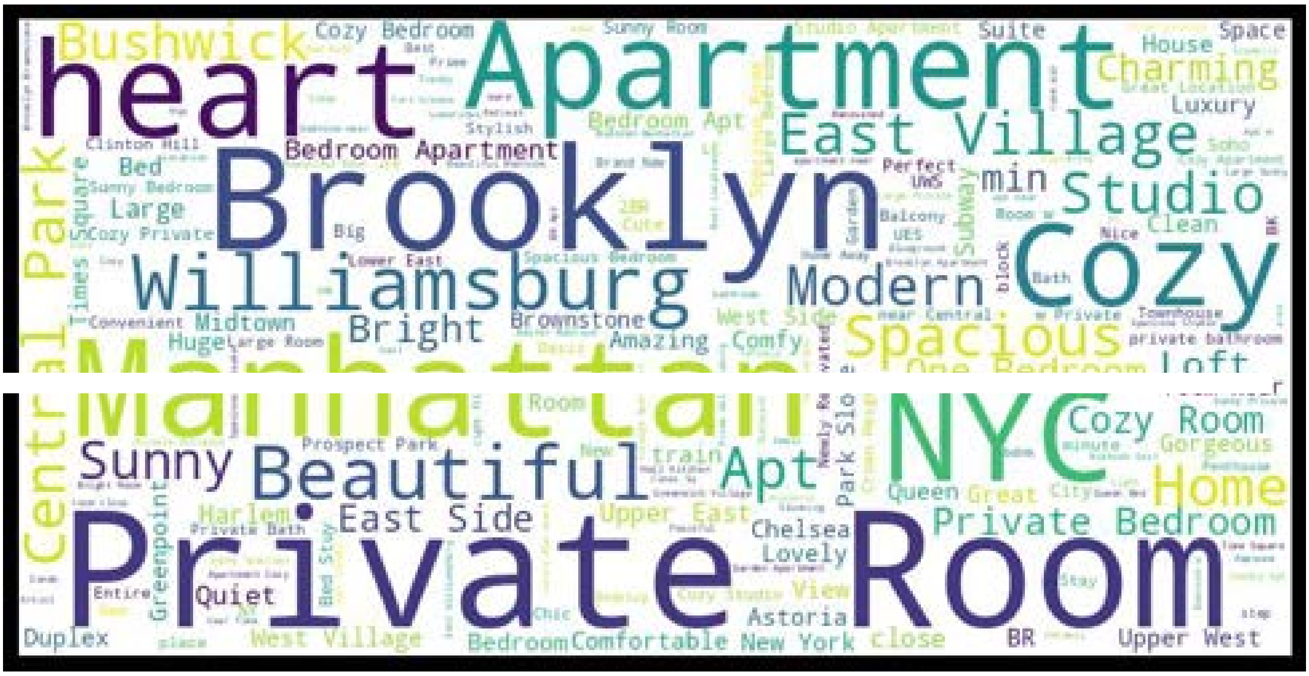
Need Improvement:

- Underperformance of Shared Rooms: The very small share of shared rooms (2.4%) may indicate a lack of interest in this room type. This could signal negative growth potential if not addressed.
- Justification: The demand for shared rooms might be limited by traveler preferences for more privacy. Additionally, shared rooms may not align well with current hospitality trends, where people value personal space and comfort. However, there might still be niche markets, such as for budget travelers or younger backpackers, that Airbnb could tap into.
- Recommendation: Airbnb could either choose to de-prioritize this segment or work on initiatives (e.g., improved features or pricing strategies) to boost demand for shared rooms.

Chart - 4 : Word Cloud Visualization : Most Common Words in Airbnb NYC Listings (2019)

```
from wordcloud import WordCloud

# Sample text data - for demonstration, assuming we are using listing names
text = ' '.join(Airbnb_df_cleaned['listing_name'])

# Generate the word cloud
wordcloud = WordCloud(width=800, height=400, background_color='white', colormap='viridis'
                      max_words=200).generate(text)

# Plot the word cloud
```

```
plt.figure(figsize=(8, 4), facecolor='black')
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```



1. Why did you pick the specific chart?

I picked a word cloud for this analysis because it provides a visual representation of the most common words used in Airbnb listings in New York City (2019). Word clouds are ideal for highlighting the most frequent terms in textual data, with the size of each word representing its frequency. This method gives a quick, intuitive overview of the dataset, making it easy to spot patterns or themes in the listing names, which may be otherwise difficult to discern in raw text format.

2. What is/are the insight(s) found from the chart?

- Neighborhood Focus: Words like "Brooklyn" and "Manhattan" are among the largest, indicating these are the most commonly mentioned neighborhoods in Airbnb listings.
- Descriptive Terms: Words such as "Private," "Room," "Beautiful," "Cozy," "Apartment," "Sunny," and "Spacious" are prominent, indicating that hosts often use these terms to describe their listings, suggesting that comfort and aesthetics are highly emphasized.
- Apartment Type: Words like "Studio," "Apt," and "Bedroom" show that many listings are focused on apartments, rooms, and studios.

- Location Highlights: Specific locations such as "Williamsburg," "East Village," and "Central Park" suggest that hosts frequently highlight well-known neighborhoods and landmarks in NYC to attract guests.

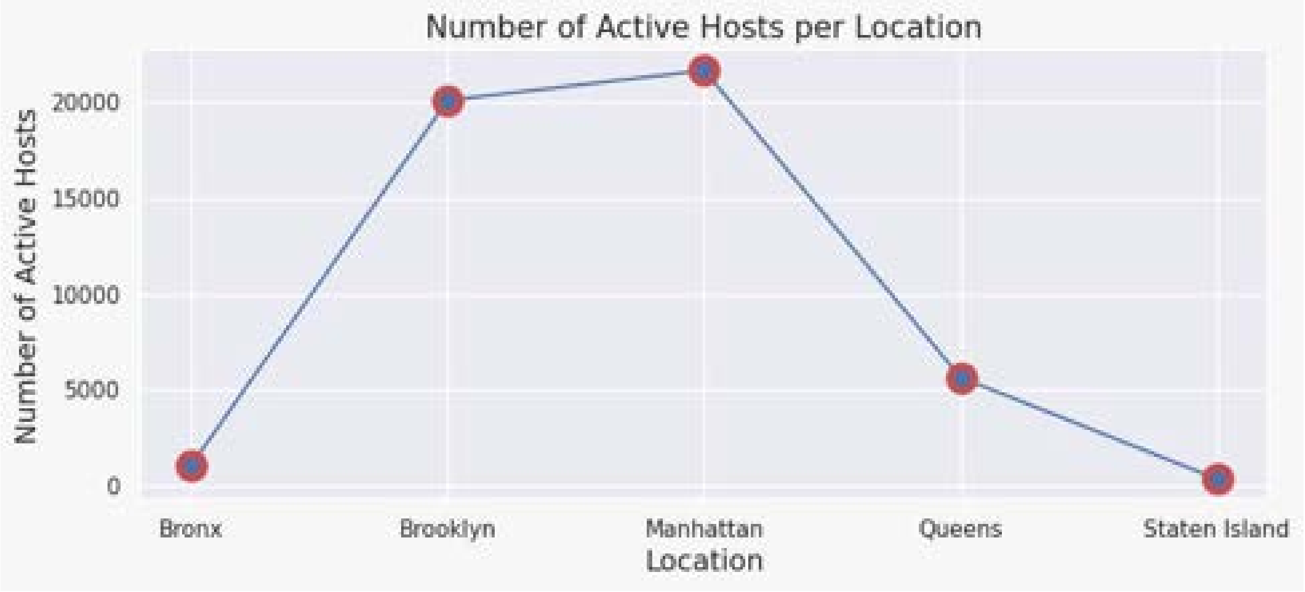
These insights reveal key aspects of Airbnb listings, such as common amenities and room types, preferred neighborhood mentions, and the overall tone used by hosts to appeal to potential guests.

Chart - 5 : Line Plot : Distribution of Active Airbnb Hosts Across NYC Boroughs

```
# Group the data by neighbourhood_group and count the number of listings for each group
hosts_per_location = Airbnb_df.groupby('neighbourhood_group')['listing_id'].count()

# Get the list of neighbourhood_group names
locations = hosts_per_location.index

# Get the list of host counts for each neighbourhood_group
host_counts = hosts_per_location.values

hosts_per_location
```

```
                          listing_id
    neighbourhood_group
          Bronx                 1091
         Brooklyn              20104
         Manhattan             21661
          Queens                5666
       Staten Island             373

  dtype: int64
```

```
# Set the figure size
plt.figure(figsize=(10, 4), facecolor='#F7F7F7')

# Create the line chart with some experiments using marker function
plt.plot(locations, host_counts, marker='o', ms=12, mew=4, mec='r')

# Add a title and labels to the x-axis and y-axis
plt.title('Number of Active Hosts per Location', fontsize='15')
plt.xlabel('Location', fontsize='14')
plt.ylabel('Number of Active Hosts', fontsize='14')

# Show the plot
plt.show()
```



1. Why did you pick the specific chart?

A line chart was chosen because:

- Visual Clarity: It clearly shows trends and comparisons of the number of active hosts across different locations. Line charts are excellent for demonstrating relationships between categories when you want to highlight increases or decreases.
- Markers: The addition of markers helps emphasize the data points for each neighborhood group, making it easier to identify the number of hosts per location.
- Sequential Ordering: The chart is effective for representing a continuous flow, showing how the number of active hosts changes between different boroughs.

2. What is/are the insight(s) found from the chart?

- Manhattan Dominates: The number of active hosts is highest in Manhattan, followed closely by Brooklyn. These two boroughs are Airbnb's key markets in NYC.
- Queens and Staten Island Lag: There is a sharp drop in the number of hosts in Queens and Staten Island, indicating lower Airbnb activity in these areas.
- Bronx Growth: The Bronx, while starting with a lower number of hosts, shows significant growth compared to Queens and Staten Island, surpassing both of these boroughs in active listings.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, these insights can have several positive business impacts:

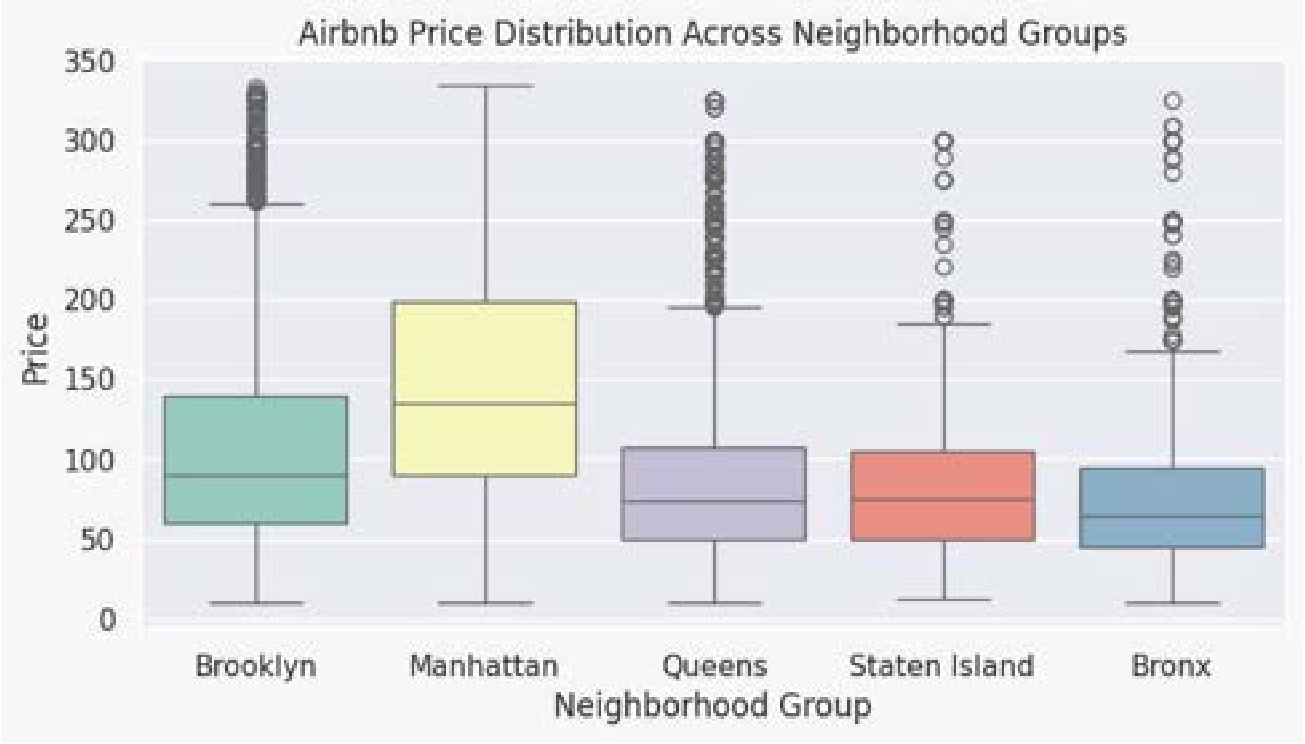
- Targeted Marketing: Knowing that Manhattan and Brooklyn are key markets, Airbnb can focus marketing efforts and business partnerships in these areas to capitalize on their popularity.
- Strategic Expansion: Bronx is emerging as a potential growth area for hosts. This insight can lead to initiatives to encourage more listings in the Bronx.
- Resource Allocation: Airbnb can optimize resources and customer support in high-traffic boroughs while identifying areas for improvement in Queens and Staten Island.
- Low Host Counts in Queens and Staten Island: The significantly lower number of active hosts in Queens and Staten Island suggests a lack of Airbnb activity, which might indicate less demand or fewer attractive properties.

Justification: This could lead to negative growth if these areas are neglected. Airbnb may want to investigate reasons for the lower activity—whether it's related to regulations, host reluctance, or simply lower demand—and implement strategies to stimulate growth, such as better marketing or incentives for new hosts.

Bivariate Analysis (2 variables)

Chart - 6 : Boxplot : Price Distribution Across Neighborhood Groups

```
# Set the figure size
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')

# Create boxplot for prices across neighborhood groups
sns.boxplot(x='neighbourhood_group', y='price', data=Airbnb_df_cleaned, palette='Set3')

# Add plot title and axis labels
plt.title('Airbnb Price Distribution Across Neighborhood Groups')
plt.xlabel('Neighborhood Group')
plt.ylabel('Price')

# Show plot
plt.show()
```



1. Why did you pick the specific chart?

I picked a boxplot chart to visualize the price distribution across neighborhood groups because it is an effective way to compare the distribution of a continuous variable (price) across different categories (neighborhood groups). Boxplots allow us to see the median, quartiles, and outliers for each group, providing a clear visual representation of the data.

2. What is/are the insight(s) found from the chart?

- The median price varies significantly across neighborhood groups, with Manhattan having the highest median price and the Bronx having the lowest.
- The price range (interquartile range) also varies across neighborhood groups, with Manhattan having the widest range and the Bronx having the narrowest.
- There are outliers in each neighborhood group, indicating that there are some listings with significantly higher or lower prices than the rest.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

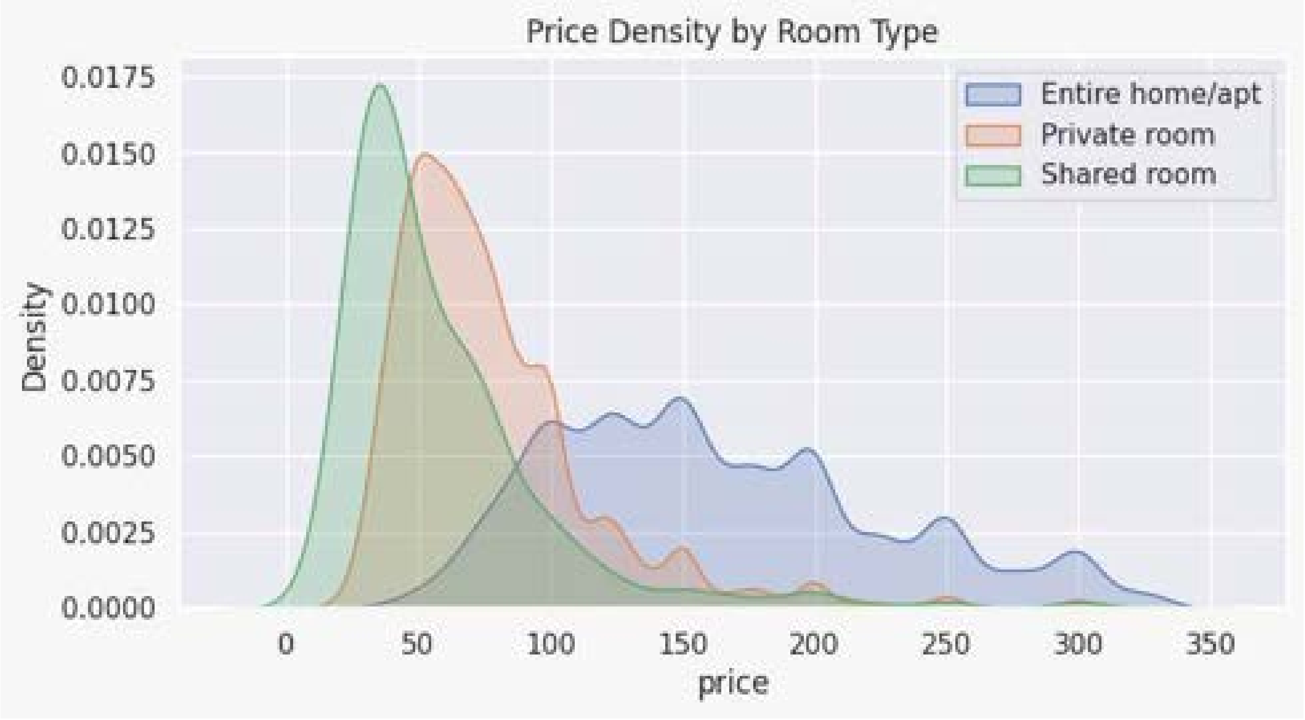
- By understanding the price distribution across neighborhood groups, Airbnb can tailor its pricing strategy to each area, ensuring that hosts are pricing their listings competitively and attracting the right type of guests.

- The insights can also help Airbnb to identify areas with potential for growth, such as neighborhoods with a high demand for listings but limited supply.
- Additionally, the insights can inform Airbnb's marketing efforts, allowing them to target specific neighborhood groups with tailored messaging and promotions.

- One potential insight that could lead to negative growth is the presence of outliers in each neighborhood group. These outliers could be listings that are significantly overpriced or underpriced compared to the rest of the market, which could lead to a negative user experience and decreased bookings. To mitigate this risk, Airbnb could consider implementing pricing guidelines or recommendations for hosts, or providing tools to help hosts optimize their pricing strategy.

Chart - 7 : Kernel Density Estimate (KDE) plot: Distribution of Airbnb Prices by Room Type in NYC

```
# Set the figure size
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')

# Plot KDE for 'Entire home/apt'
sns.kdeplot(
    Airbnb_df_cleaned[Airbnb_df_cleaned['room_type'] == 'Entire home/apt']['price'],
    label='Entire home/apt',
    shade=True
)

# Plot KDE for 'Private room'
sns.kdeplot(
    Airbnb_df_cleaned[Airbnb_df_cleaned['room_type'] == 'Private room']['price'],
    label='Private room',
    shade=True
)

# Plot KDE for 'Shared room'
sns.kdeplot(
    Airbnb_df_cleaned[Airbnb_df_cleaned['room_type'] == 'Shared room']['price'],
    label='Shared room',
    shade=True
)

# Add title and legend
plt.title('Price Density by Room Type')
plt.legend()

# Display the plot
plt.show()
```



1. Why did you pick the specific chart?

The Kernel Density Estimate (KDE) plot is ideal for visualizing the distribution of continuous variables, such as prices. It provides a smooth curve that helps identify trends, patterns, and the overall shape of the data for different room types. In this case, comparing the prices for different Airbnb room types (Entire home/apt, Private room, and Shared room) helps reveal price disparities, patterns, and possible price ranges.

2. What is/are the insight(s) found from the chart?

- Entire home/apt: The price distribution for entire homes/apartments is spread out, with a peak around dollar 100. The prices range widely, even going above dollar 250.
- Private room: The distribution is concentrated between dollar 50 and dollar 100, with a higher peak at lower prices compared to entire homes/apartments.
- Shared room: The prices for shared rooms are very low, with most prices clustered below dollar 50.

Overall, the chart shows that shared rooms are the cheapest, followed by private rooms, with entire homes/apartments being the most expensive. There is a clear distinction in price distribution across room types.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

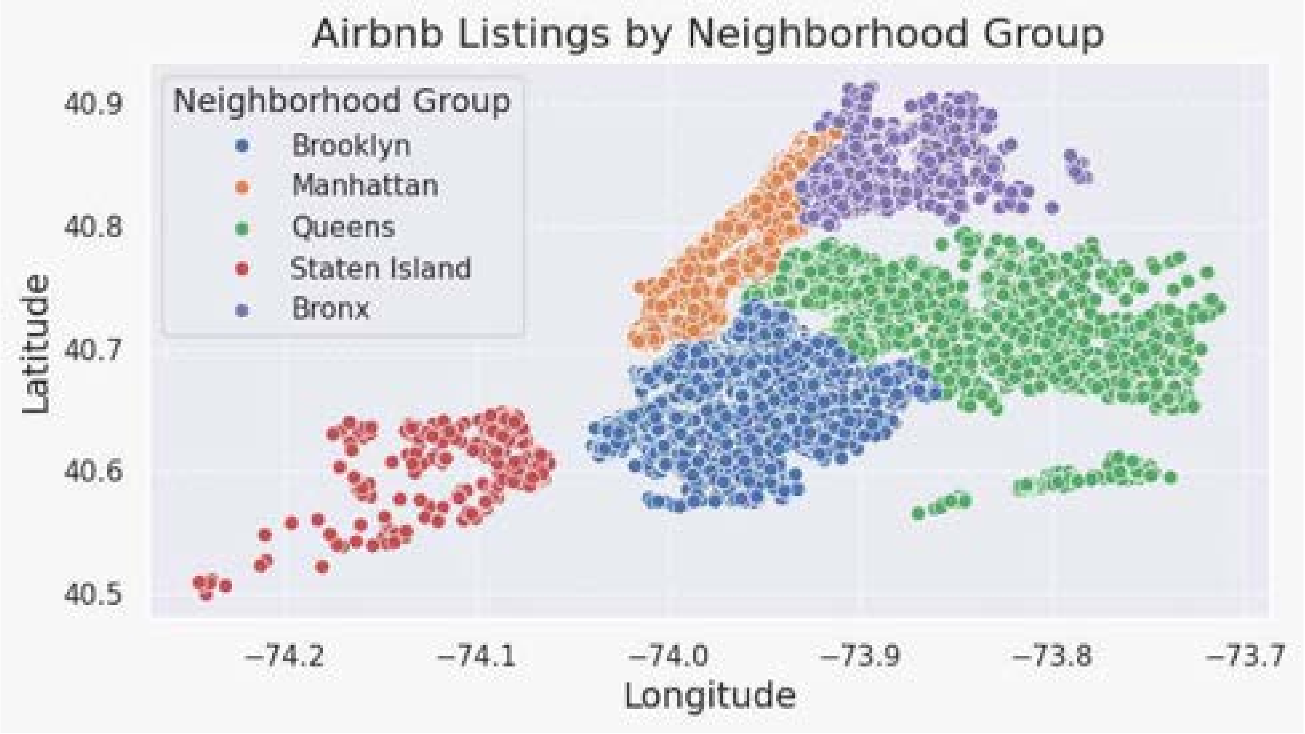
Yes, these insights are valuable for multiple stakeholders:

- Hosts: Hosts can adjust their pricing strategy based on the room type and local market trends. If a host is offering an entire home, they can see that there is a range of prices, and pricing around dollar 100 could capture a large part of the demand. Private room hosts might aim to price competitively around dollar 50 – dollar 100 to attract more guests.
- Airbnb: Airbnb could use this information to promote certain room types depending on the target market or to suggest optimal pricing for hosts to maximize occupancy and revenue.
- Travelers: This can help travelers find the most affordable options based on their accommodation preferences.

One potential insight that could be seen as negative is the narrow price range and low peaks for shared rooms. This suggests that shared rooms may not be as profitable for hosts. If too many hosts list shared rooms at very low prices, it may reduce profitability and lead to less host engagement in offering such listings, which could result in fewer available budget-friendly options. Additionally, for hosts who primarily list shared rooms, the narrow price range could limit their ability to differentiate or increase their earnings over time.

Chart - 8 : Scatterplot : Geographic Distribution of Airbnb Listings by Neighborhood Group in NYC

```
# Create a scatter plot to visualize Airbnb listings by neighborhood group
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')  # Set the figure size for better visibil

# Scatter plot with longitude on the x-axis and latitude on the y-axis
# Color points by 'neighbourhood_group' to differentiate between neighborhoods
sns.scatterplot(x='longitude', y='latitude', hue='neighbourhood_group',
                data=Airbnb_df_cleaned,
               )  # Remove edge color for a cleaner look

# Title and labels
plt.title('Airbnb Listings by Neighborhood Group', fontsize=16)  # Title with larger font
plt.xlabel('Longitude', fontsize=14)  # X-axis label
plt.ylabel('Latitude', fontsize=14)  # Y-axis label

# Add a legend to identify neighborhood groups
plt.legend(title='Neighborhood Group', title_fontsize='13', fontsize='11')

# Optional: Add grid for better readability
plt.grid(True, linestyle='--', alpha=0.6)

# Show the plot
plt.show()
```



1. Why did you pick the specific chart?

I picked a scatterplot to visualize the geographic distribution of Airbnb listings by neighborhood group in NYC because it is an effective way to show the spatial relationships between listings and neighborhood groups. Scatterplots allow us to see the density and distribution of listings across different areas, providing a clear visual representation of the data.

2. What is/are the insight(s) found from the chart?

- Manhattan has the highest concentration of listings, forming a dense cluster.
- Brooklyn has a large number of listings spread over a wider area.
- Queens shows listings distributed across multiple distinct clusters.
- Staten Island has the fewest listings, concentrated in the north of the borough.
- The Bronx has a moderate number of listings, mostly in the south.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

1. These insights can create positive business impact:

- They can help Airbnb identify areas for growth or market saturation.
- Hosts can use this information to strategically price their listings based on location.

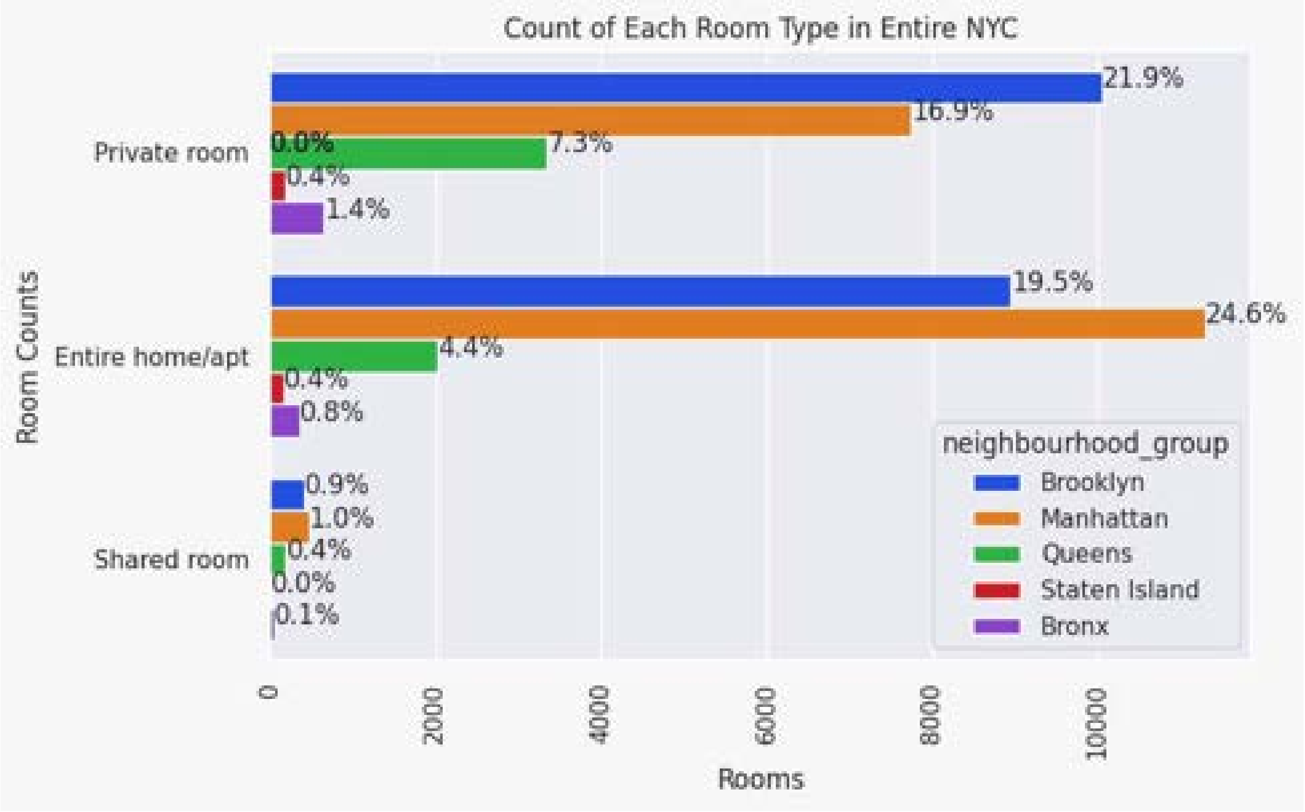
- Airbnb can target marketing efforts in underserved areas.

2. Potential insights leading to negative growth:

- Over-concentration in Manhattan might lead to increased scrutiny from regulators.
- Low listing density in some areas (e.g., Staten Island) might indicate less tourism appeal or infrastructure, potentially limiting growth.
- Clustering of listings in certain neighborhoods could lead to community backlash or zoning issues, potentially restricting future growth in those areas.

Chart - 9 : Stacked Horizontal Bar Plot : Distribution of accommodation types across New York City's

```
# Set the size of the plot (8 inches by 5 inches)
plt.figure(figsize= (8, 5), facecolor='#F7F7F7')

# Create a count plot with room_type on the y-axis, grouped by neighbourhood_group using
ax = sns.countplot(y='room_type', hue='neighbourhood_group', data=Airbnb_df_cleaned, pale

# Get the total number of room_type listings to calculate percentages
total = len(Airbnb_df_cleaned['room_type'])

# Loop over each bar in the plot and annotate it with the corresponding percentage
for p in ax.patches:
  # Calculate the percentage for each bar (width of the bar divided by the total number o
  percentage = '{:.1f}%'.format(100 * p.get_width() / total)

  # Set the x position for the annotation slightly beyond the bar's end
  x = p.get_x() + p.get_width() + 0.0

  # Set the y position to the middle of the bar
  y = p.get_y() + p.get_height() / 2

  # Annotate the plot with the calculated percentage at the specified (x, y) position
  ax.annotate(percentage, (x, y))

# Add a title to the plot
plt.title('Count of Each Room Type in Entire NYC')

# Label the x-axis
plt.xlabel('Rooms')

# Rotate the x-axis tick labels by 90 degrees for readability
plt.xticks(rotation=90)

# Label the y-axis
plt.ylabel('Room Counts')

# Display the final plot with the annotations and labels
plt.show()
```



1. Why did you pick the specific chart?

I picked a **stacked horizontal bar** chart because it is highly effective for displaying the count of listings for each room type across different neighborhood groups. The hue feature clearly distinguishes between the neighborhood groups, while the horizontal bars allow easy comparison of room types (like Private room, Entire home/apt, and Shared room). This chart is intuitive and provides a clear breakdown of the distribution of room types in each neighborhood group.

2. What is/are the insight(s) found from the chart?

The key insights from the chart are:

- Entire home/apartment listings are most common in Manhattan (24.6%) and Brooklyn (19.5%), which aligns with the popularity of these neighborhoods for full-property rentals.
- Private rooms have a significant presence in Brooklyn (21.9%) and Manhattan (16.9%), indicating that many hosts offer single rooms, especially in these boroughs.
- Shared rooms are rare in all neighborhoods, making up a very small fraction of the total listings, suggesting less demand for shared spaces.

- Queens has a relatively lower percentage of listings for both room types compared to Brooklyn and Manhattan.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, these insights can be valuable for different stakeholders:

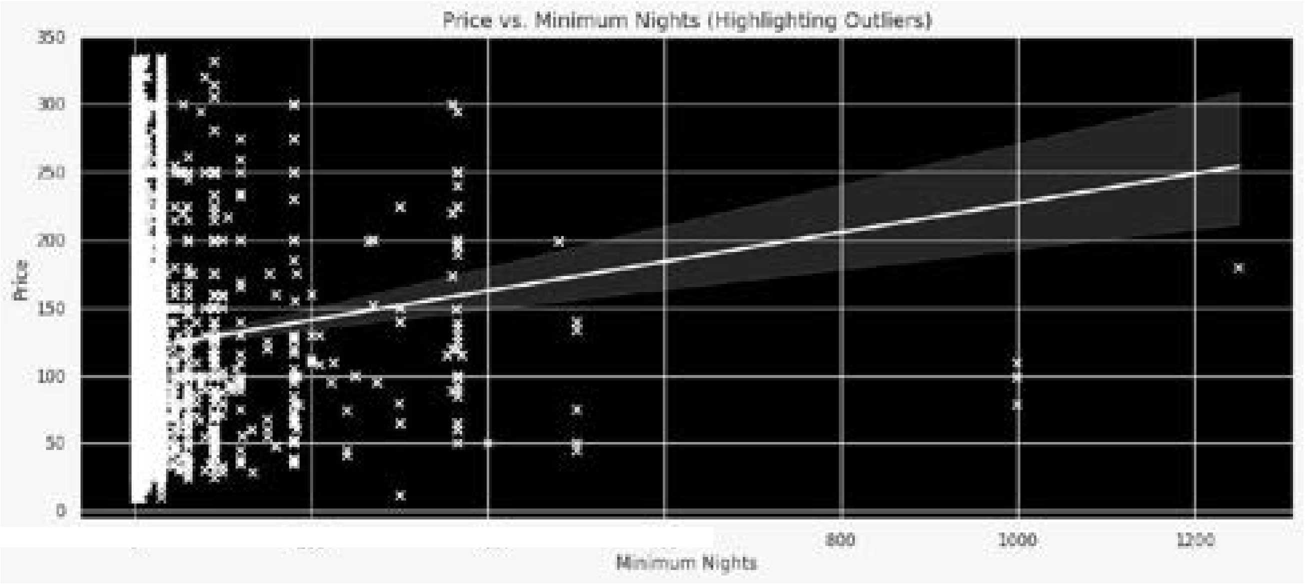
- **Airbnb hosts** can use this information to tailor their offerings based on neighborhood demand. For instance, hosts in Brooklyn might consider listing more private rooms, as they are in demand, while in Manhattan, focusing on entire apartments might generate higher revenues.
- **Travelers** can gain insights into the type of accommodation available in each neighborhood and use it to plan their stays better, depending on their preferences and budget.
- **Airbnb's Business Strategy**: The company can use this data to drive targeted marketing efforts, especially in boroughs with lower listing counts (like Staten Island), where they might promote host sign-ups to expand inventory.

There are no direct signs of negative growth based on this chart. However, the small share of shared rooms across all neighborhood groups might be a concern for hosts looking to offer such accommodations. This low share suggests that there is low demand for shared spaces, meaning hosts may face challenges filling such listings. It could also indicate that customers prefer more privacy (entire apartments or private rooms), and focusing on shared rooms may not yield much return on investment. Airbnb may need to reconsider its promotion or pricing strategies for shared rooms.

Chart - 10 : Scatter Plot with Regression Line (Including Outliers) : Impact of Minimum Stay Duration on Airbnb Pricing

```
# Scatter plot with regression line and custom marker
plt.figure(figsize=(15, 6), facecolor='#F7F7F7')

# Set the face color of the figure (axes) to black
ax = plt.gca()  # Get current axes
ax.set_facecolor('black')  # Set background color of the plot to black

sns.regplot(x='minimum_nights', y='price', data=Airbnb_df_cleaned, marker = "x", scatter_

# Highlighting the title and axis labels
plt.title('Price vs. Minimum Nights (Highlighting Outliers)', fontsize=14)
plt.xlabel('Minimum Nights', fontsize=12)
plt.ylabel('Price', fontsize=12)
```

```
# Add grid and display plot
plt.grid(True)
plt.show()
```



1. Why did you pick the specific chart?

Chart selection rationale: A scatter plot with a regression line was chosen because it effectively visualizes the relationship between two continuous variables (minimum nights and price) while also showing the overall trend and potential outliers.

2. What is/are the insight(s) found from the chart?

Insights from the chart:

- There's a slight positive correlation between minimum nights and price.
- Most listings have short minimum stay requirements (clustered on the left).
- There's high price variability for shorter minimum stays.
- Some outliers exist with very high prices or extended minimum stay requirements.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Business impact of insights:

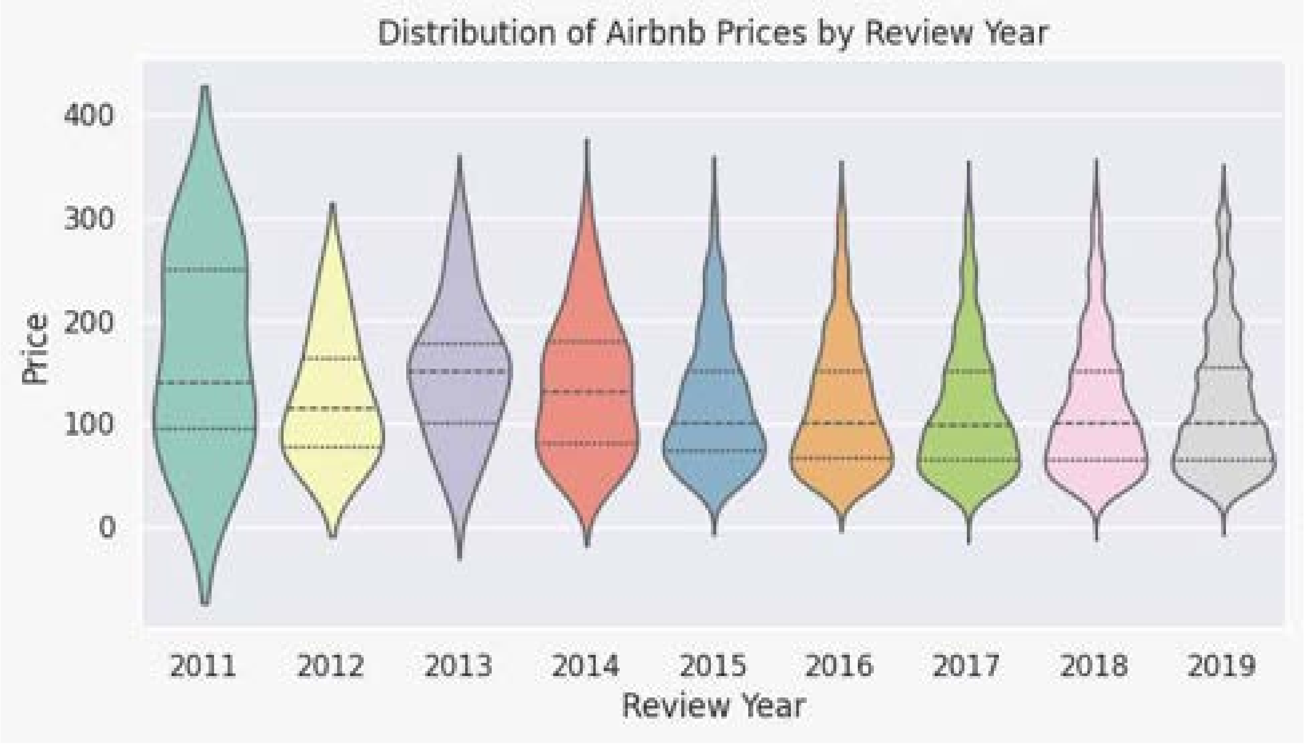
- Positive impact potential: Understanding the relationship between minimum stay and pricing can help hosts optimize their pricing strategies and minimum night requirements to maximize occupancy and revenue.
- The data suggests flexibility in pricing for shorter stays, which could be leveraged for dynamic pricing models.

Insights leading to potential negative growth:

- The wide scatter of prices for shorter minimum stays suggests high competition in this segment, which could lead to price wars and reduced profitability if not managed carefully.
- The presence of outliers with very high minimum night requirements might indicate properties that struggle to attract bookings, potentially leading to reduced overall platform activity if this becomes a trend.

Chart - 11 : Violin Plot : Airbnb Prices by Review Year

```
# Filter out rows where the review year is 1970 (Impute missing data: Impute the missing
Airbnb_df_cleaned = Airbnb_df_cleaned[Airbnb_df_cleaned['review_year'] != 1970]

# Replot the chart after cleaning
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')
sns.violinplot(x='review_year', y='price', data=Airbnb_df_cleaned, palette='Set3', inner=
plt.title('Distribution of Airbnb Prices by Review Year')
plt.xlabel('Review Year')
plt.ylabel('Price')
plt.show()
```



1. Why did you pick the specific chart?

The violin plot was selected because it provides a clear visual representation of the distribution of Airbnb prices across different review years. The plot combines the features of both a box plot and a kernel density estimate, showing the spread and density of the data, while also indicating the quartiles and medians for each year. This makes it easier to observe:

- The variability of prices within each year.
- Any trends over time, including changes in the distribution of prices and outliers.

Violin plots are particularly useful when examining the distribution of numerical data over categories like years.

2. What is/are the insight(s) found from the chart?

- **Price Volatility**: Some years show more volatile price distributions (wider violins), indicating that prices were more spread out.
- **Price Peaks and Drops**: Certain years like 2011 and 2013 have higher median prices, suggesting price peaks in those periods. In contrast, years like 2019 show a narrower distribution with lower median prices.
- **Stable Periods**: Years such as 2014 and 2015 have relatively consistent price distributions, with more tightly grouped price ranges.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, the insights can lead to positive business impacts:

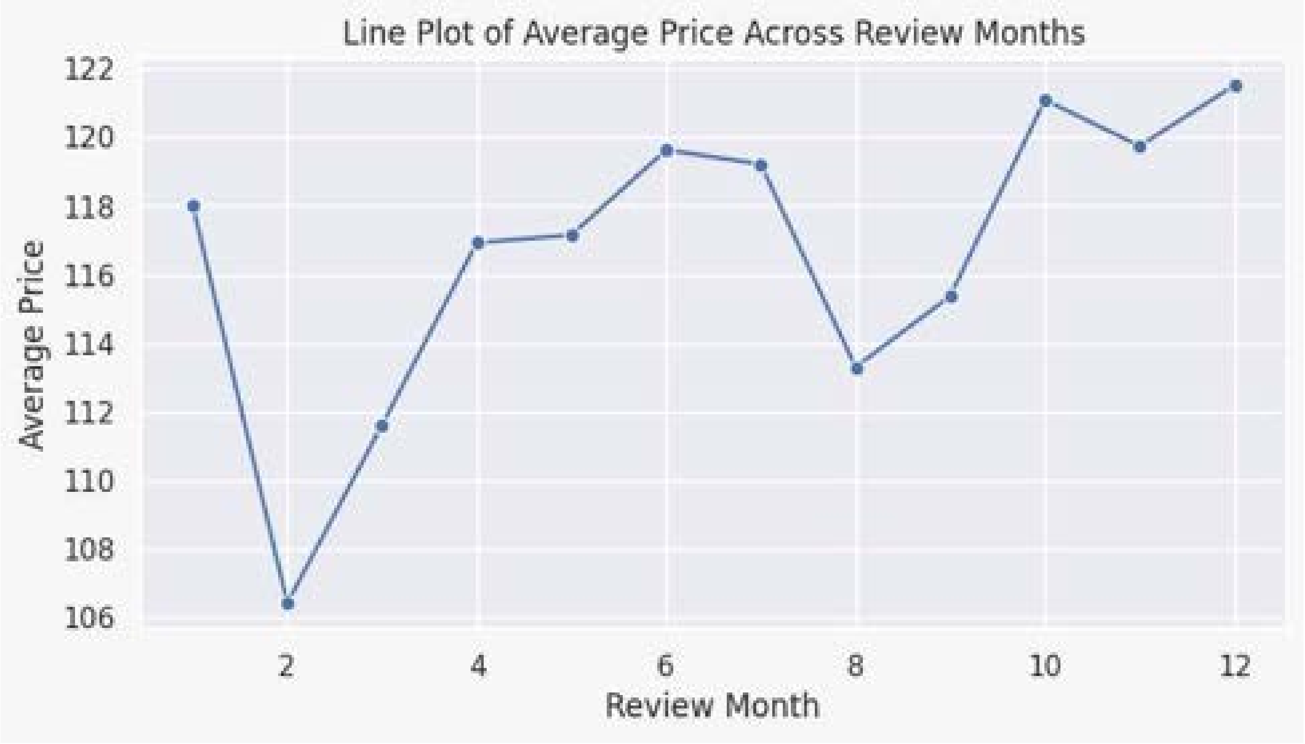
- **Pricing Strategy**: Understanding the price trends over different review years can help Airbnb hosts and platform managers adjust their pricing strategies. For example, if certain years had higher prices, hosts can plan their pricing for upcoming years based on similar trends.
- **Trend Analysis**: By identifying years where prices peaked or dropped, Airbnb can adjust marketing efforts during periods of high demand or plan special offers to boost sales in slower periods.

Negative Analysis,

- **Price Decline**: If prices are seen to decline consistently over the years, especially towards 2019, this could signal reduced demand for Airbnb listings, leading to a potential negative impact on growth.
- **Market Saturation**: If the distribution becomes narrower and prices stabilize or drop over time, it could suggest market saturation, where increased competition drives prices down. This could lead to negative growth for hosts who rely on higher price points for profitability.
- **Economic Challenges**: If price drops correspond with specific years, it could indicate economic downturns, impacting the platform's revenue and hosts' earnings.

Chart - 12 : Line Plot : Distribution of Average Price Across Review Months

```
# Group data by review_month and calculate the mean price
average_price_per_month = Airbnb_df_cleaned.groupby('review_month')['price'].mean().reset

# Create a line plot to visualize the average price per month
plt.figure(figsize=(8, 4), facecolor='#F7F7F7')
sns.lineplot(x='review_month', y='price', data=average_price_per_month, marker='o', color

# Setting the title and labels
plt.title('Line Plot of Average Price Across Review Months')
plt.xlabel('Review Month')
plt.ylabel('Average Price')

# Display the plot
plt.show()
```



1. Why did you pick the specific chart?

I chose the line plot because it is ideal for displaying trends over time or sequences, which aligns with the goal of visualizing how average Airbnb prices change across different months. Since we are analyzing the trend of price based on review_month (a time-based categorical variable), the line plot is well-suited to showing both the progression and the seasonal pattern. The markers ('o') help highlight individual points, making it easier to observe monthly fluctuations.

2. What is/are the insight(s) found from the chart?

From the line plot, we can identify several insights:

- There may be some months where prices spike, indicating potential seasonal demand for Airbnb listings.
- If there are certain months where the price dips, it might suggest lower demand or more availability in those months, leading to more competitive pricing.
- The overall price trend over months can show whether there are consistent patterns, such as prices being higher in the summer months (typically June to August) or lower in the off-season.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, the insights can significantly help in business decision-making:

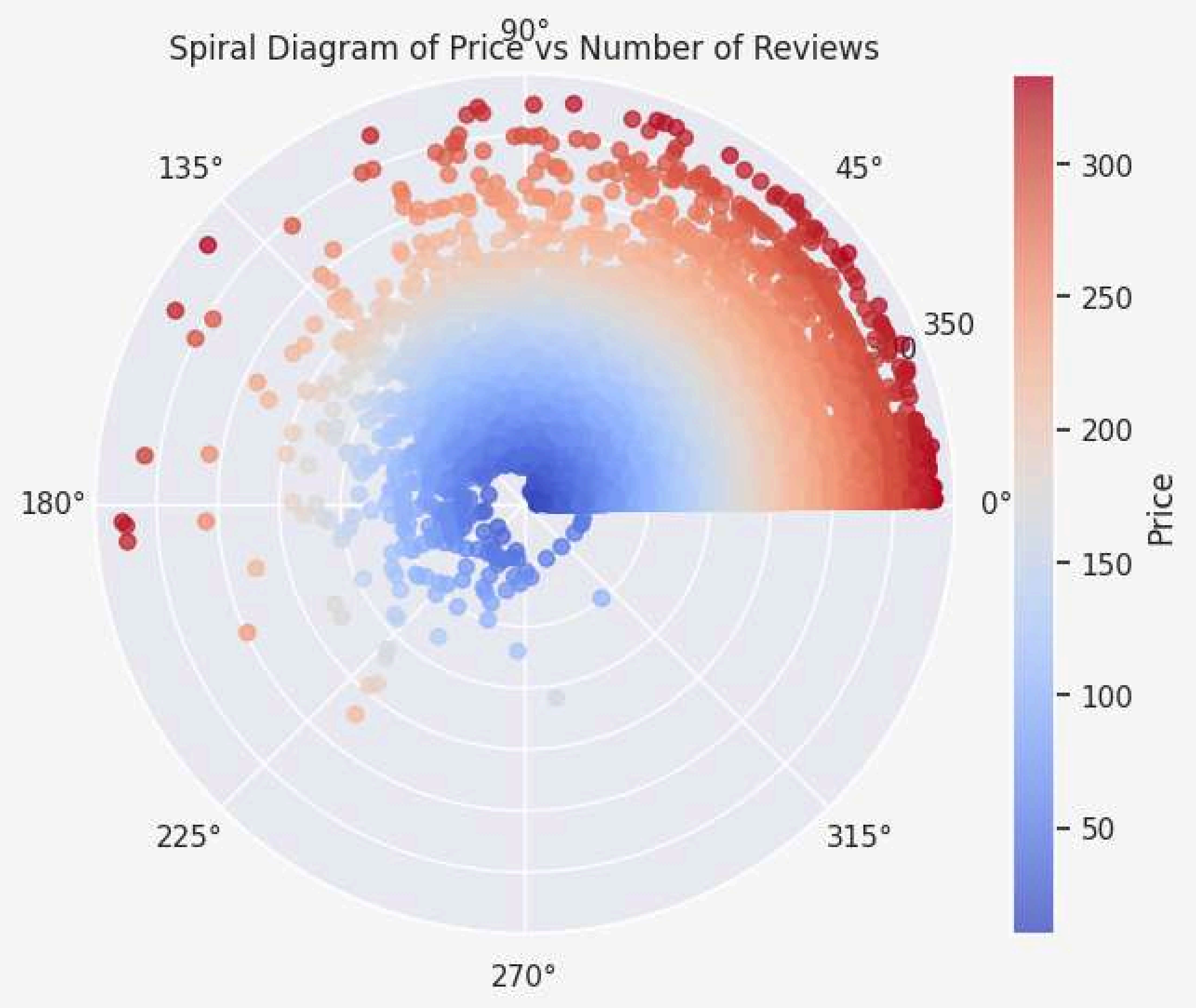
- **Pricing Strategy**: Hosts can adjust their pricing based on the seasonality. For instance, if certain months have higher demand (and thus higher prices), they can raise prices to maximize profit during those months.
- **Revenue Forecasting**: By knowing the months with higher or lower prices, hosts and Airbnb can better predict revenue throughout the year.
- **Targeted Promotions**: During months where prices dip, Airbnb could launch targeted promotions or marketing campaigns to boost occupancy rates.

There are potential negative growth indicators that could emerge from this analysis:

- **Price Volatility**: If the chart shows large fluctuations in prices between months, it may suggest inconsistency in demand. This unpredictability could make it difficult for hosts to maintain stable earnings.
- **Low Price Periods**: Months with significantly lower average prices may point to low demand periods, which could hurt overall profitability. To counteract this, hosts might need to offer discounts or special deals, potentially reducing their margins during these months.

Chart - 13 : Spiral Diagram : Relationship Between The Price And The Total Reviews

```
# Create a new column for the number of reviews
Airbnb_df_cleaned['num_reviews'] = Airbnb_df_cleaned['total_reviews']

# Convert the number of reviews to radians for polar plotting
Airbnb_df_cleaned['num_reviews_rad'] = Airbnb_df_cleaned['num_reviews'] * (2 * np.pi / Ai

# Set up the polar plot
plt.figure(figsize=(8, 6), facecolor='#F7F7F7')
ax = plt.subplot(111, polar=True)

# Create a scatter plot on the polar axis
sc = ax.scatter(Airbnb_df_cleaned['num_reviews_rad'], Airbnb_df_cleaned['price'], c=Airbn

# Add color bar for price scale
plt.colorbar(sc, label='Price')

# Set the title
plt.title('Spiral Diagram of Price vs Number of Reviews')

# Show the plot
plt.show()
```



1. Why did you pick the specific chart?

I picked the spiral diagram because it effectively visualizes the relationship between two variables (price and number of reviews) in a compact and visually striking format. The spiral shape allows for a clear representation of the correlation between the two variables, making it easy to identify patterns and trends.

2. What is/are the insight(s) found from the chart?

From the chart, we can see that:

- There is a positive correlation between price and number of reviews. As the number of reviews increases, the price tends to increase as well.
- The highest concentration of reviews is between 200-350 reviews, which corresponds to the highest prices.
- There is a notable spread in prices for items with fewer reviews, suggesting more price variability for less-reviewed items.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, the gained insights can help create a positive business impact. For example:

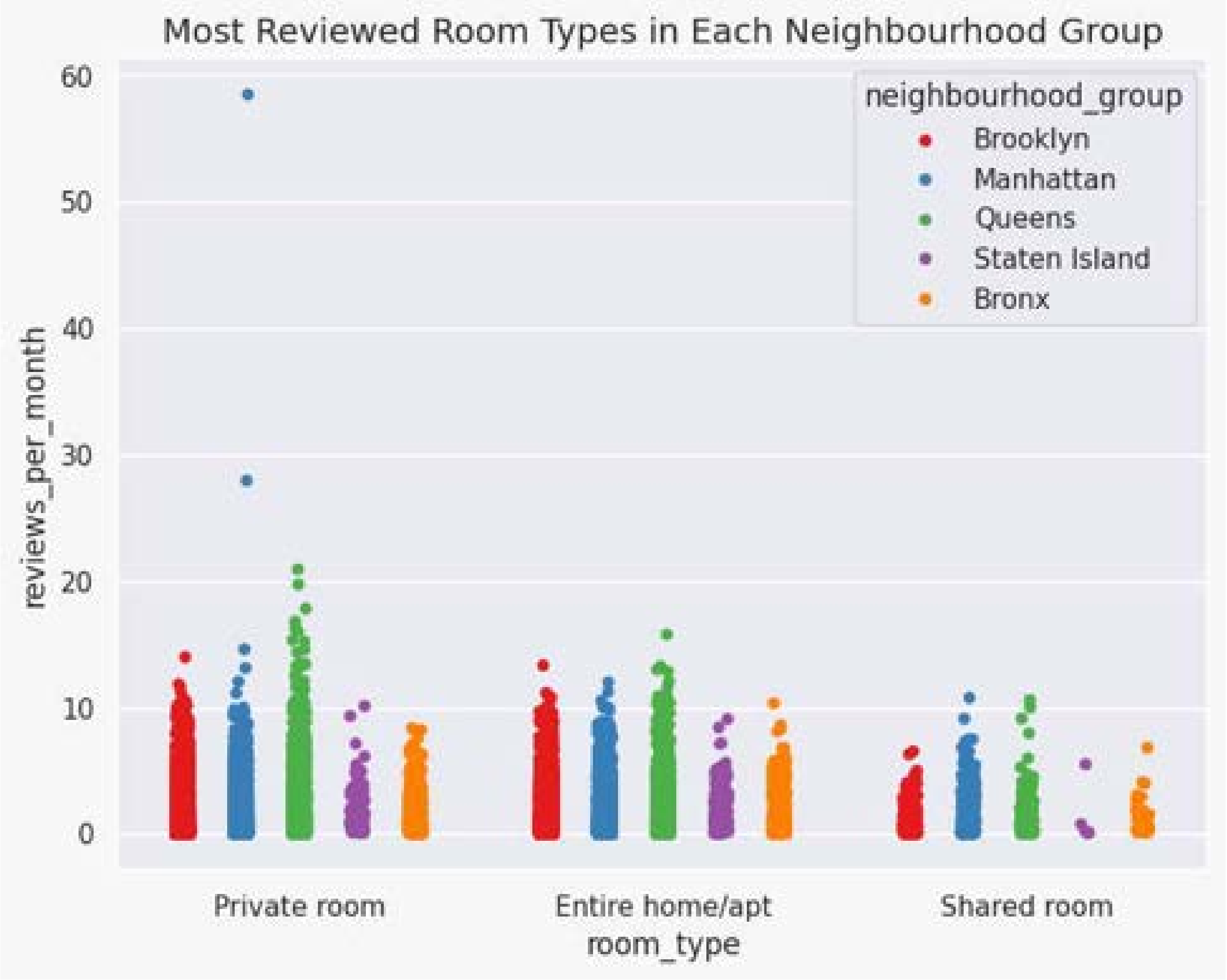
- The positive correlation between price and number of reviews suggests that accumulating reviews can lead to higher prices, which can increase revenue.
- The concentration of high-priced items with many reviews suggests that focusing on quality to gain positive reviews can allow for higher pricing.
- The spread in prices for less-reviewed items suggests that pricing new or less-reviewed items more competitively can help gain traction.

No, there don't appear to be any insights that lead to negative growth. However, the lack of high-priced items with few reviews might indicate a challenge in introducing new premium products without an established review base. This could be addressed through targeted marketing or review incentive programs for new high-end offerings.

Chart - 14 : Strip Plot : Relationship between Monthly Reviews and Room Types in each Neighborhood group

```
# Create a figure and a set of subplots with specified size
f, ax = plt.subplots(figsize=(8, 6), facecolor='#F7F7F7')

# Plot a strip plot
ax = sns.stripplot(x='room_type', y='reviews_per_month', hue='neighbourhood_group', dodge

# Set the title for the plot
ax.set_title('Most Reviewed Room Types in Each Neighbourhood Group', fontsize='14')
```

```
Text(0.5, 1.0, 'Most Reviewed Room Types in Each Neighbourhood Group')
```



1. Why did you pick the specific chart?

- Visual Clarity: Strip plots are effective for visualizing the distribution of a numerical variable (reviews_per_month) across different categories (room_type), especially when dealing with overlapping data points.
- Categorical Comparison: It allows for easy comparison of the distribution of review counts for different room types across various neighborhood groups, helping to identify patterns or discrepancies.
- Highlighting Differences: Using hue for neighborhood groups and dodge=True helps differentiate between groups and reduces clutter, making it easier to analyze how review counts vary by room type and neighborhood group.

2. What is/are the insight(s) found from the chart?

We can see that Private room recieved the most no of reviews/month where Manhattan had the highest reviews received for Private rooms with more than 50 reviews/month, followed by Manhattan in the chase.

Manhattan & Queens got the most no of reviews for Entire home/apt room type.

There were less reviews recieved from shared rooms as compared to other room types and it was from Staten Island followed by Bronx.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, the insights can positively impact the business in several ways:

- Targeted Marketing: Understanding which room types receive more reviews in certain neighborhoods can help in tailoring marketing strategies. For instance, focusing promotions on popular room types in high-review neighborhoods.
- Improving Offerings: Identifying high-review room types can guide Airbnb hosts to focus on or enhance similar offerings, potentially increasing their booking rates.
- Pricing Strategies: Insights into review trends by neighborhood can help in adjusting pricing strategies based on demand and popularity.

Potential negative insights could include:

- Low Review Counts for Certain Room Types: If the chart shows that specific room types consistently receive lower reviews across all neighborhoods, it might indicate issues with those room types, such as less appealing amenities or less desirable locations.
- Neighborhood Discrepancies: If some neighborhoods show very low review counts for popular room types, it could signal problems like poor visibility or lower demand in those areas, potentially leading to decreased bookings.

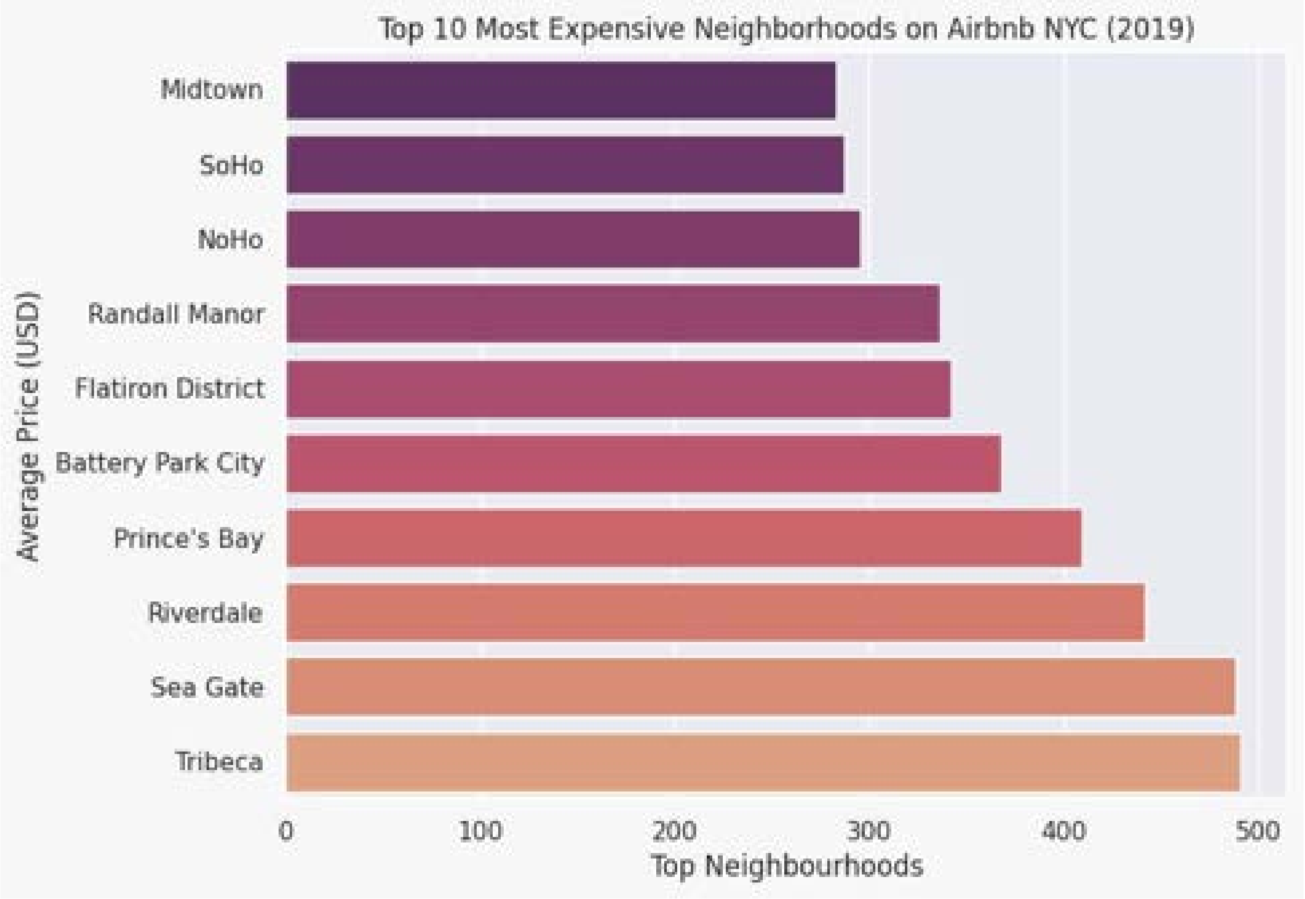
Chart - 15 : Bar Plot : Top 10 Most Expensive Neighborhoods with Price Comparison

```
# Exclude rows where the price is zero
Airbnb_df_exc = Airbnb_df[Airbnb_df['price'] != 0]

# Group by neighbourhood and calculate the mean price for each neighbourhood
neighborhood_prices = Airbnb_df_exc.groupby('neighbourhood')['price'].mean().reset_index(

# Sort by price in descending order and select the top 10 most expensive neighborhoods
top_10_expensive = neighborhood_prices.sort_values(by='price', ascending=False)

# drop 2 columns from top_10_expensive, they have single neighborhood
```

```
top_10_expensive_neighborhoods = (top_10_expensive.iloc[2:]).head(10)

# Print the results
print(top_10_expensive_neighborhoods)
```

```
          neighbourhood       price
197             Tribeca  490.638418
174            Sea Gate  487.857143
167           Riverdale  442.090909
157        Prince's Bay  409.500000
6      Battery Park City  367.557143
75     Flatiron District  341.925000
161       Randall Manor  336.000000
144                NoHo  295.717949
178                SoHo  287.103352
127             Midtown  282.719094
```

```
# Plot the top 10 most expensive neighborhoods
plt.figure(figsize=(8, 6), facecolor='#F7F7F7')  # Set the figure size and background col

# Create a horizontal bar plot with Seaborn, using the 'flare' color palette
sns.barplot(data=top_10_expensive_neighborhoods, x='price', y='neighbourhood', palette='f

# Set the title of the chart
plt.title('Top 10 Most Expensive Neighborhoods on Airbnb NYC (2019)')

# Label the x-axis and y-axis
plt.xlabel('Top Neighbourhoods')  # Label for the x-axis (represents neighborhoods)
plt.ylabel('Average Price (USD)')  # Label for the y-axis (represents average price)

# Invert the y-axis to show the most expensive neighborhood at the top
plt.gca().invert_yaxis()

# Display the plot
plt.show()
```



1. Why did you pick the specific chart?

Bar charts are ideal for comparing categories, such as neighborhood names, against a numerical value like average price. By inverting the y-axis, the chart shows the most expensive neighborhoods at the top, making it easy for viewers to grasp the key insights at a glance.

2. What is/are the insight(s) found from the chart?

The insights from the chart include:

- Most expensive neighborhoods: The chart shows *Tribeca* & *Sea Gate* neighborhood have the highest average price. It is helping to identify areas that are more lucrative for hosts or more expensive for visitors.
- Price variation: It highlights the price disparity between neighborhoods, revealing how the location significantly affects the price.
- Premium locations: It showcases where Airbnb listings command premium prices, indicating potentially high-demand or luxury locations.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, the insights can lead to positive business impacts:

- Hosts can optimize their pricing strategies by knowing the average prices in expensive neighborhoods. This can help them adjust their rates for higher profitability if they are in competitive areas.
- Airbnb can use this information to guide travelers on where to find affordable or premium stays based on their budget preferences.
- Airbnb could target marketing campaigns for high-demand, expensive neighborhoods to upscale customers or expand to nearby neighborhoods with lower prices but similar amenities.
- Real estate investors may use the data to invest in areas where Airbnb yields high returns, thus creating more inventory in those neighborhoods.

While the chart itself provides valuable insights, there are some potential risks:

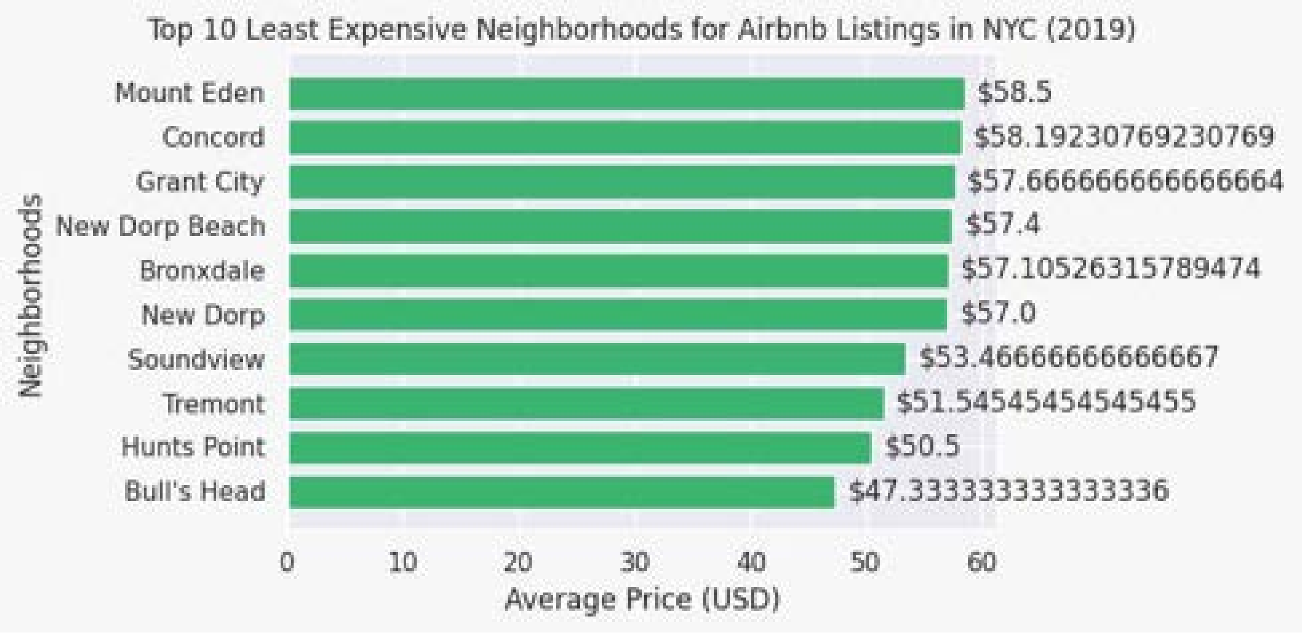
- If hosts or property managers increase their prices too aggressively based on the insights from this chart, it could deter budget-conscious travelers. High prices in already expensive neighborhoods may reduce bookings or occupancy rates, especially if alternative accommodation options are available in nearby areas.
- Some neighborhoods might reach a price saturation point where further price hikes could lead to reduced demand, leading to negative growth for hosts operating in these areas.
- As more hosts become aware of the high price potential in these neighborhoods, the market may become oversaturated, reducing profitability for individual hosts and leading to a "race to the bottom" in pricing.

Chart - 16 : Horizontal Bar Chart : Top 10 Least Expensive Neighborhoods with Price Comparison

```
# Sort by price in descending order and select the top 10 least expensive neighborhoods
cheap_10_neighbourhood = neighborhood_prices.sort_values(by='price', ascending=False, ign

# Extract 'neighbourhood' and 'price' columns into separate variables
neighbourhoods = cheap_10_neighbourhood['neighbourhood'].tolist()
prices = cheap_10_neighbourhood['price'].tolist()

# Print the results to check
print("Neighbourhoods:", neighbourhoods)
print("Prices:", prices)
```

```
Neighbourhoods: ['Mount Eden', 'Concord', 'Grant City', 'New Dorp Beach', 'Bronxdale'
Prices: [58.5, 58.19230769230769, 57.666666666666664, 57.4, 57.10526315789474, 57.0,
```

```
# Set the figure size and background color for the plot
plt.figure(figsize=(8,4), facecolor='#F7F7F7')
plt.barh(neighbourhoods, prices, color='mediumseagreen')

# Add titles and labels
plt.title('Top 10 Least Expensive Neighborhoods for Airbnb Listings in NYC (2019)')
plt.xlabel('Average Price (USD)')
plt.ylabel('Neighborhoods')

# Reverse the order of neighborhoods to show the least expensive at the top
plt.gca().invert_yaxis()

# Show the price on the bars
for index, value in enumerate(prices):
    plt.text(value + 1, index, f'${value}', va='center')

# Display the chart
plt.tight_layout()
plt.show()
```



1. Why did you pick the specific chart?

I chose a horizontal bar chart because it is an effective way to visualize a comparison of values, particularly when the categories (neighborhoods) have longer names. The horizontal orientation allows for clearer and more readable labeling, and the bars make it easy to visually compare prices across different neighborhoods. The descending order from top to bottom highlights the least expensive neighborhood at the top, making the insights intuitive to grasp.

2. What is/are the insight(s) found from the chart?

From the chart, the insights are:

- Bull's Head offers the least expensive Airbnb listings with an average price of dollar 47.33.
- Mount Eden is the most expensive among the least expensive neighborhoods, with an average price of dollar 58.50.

Other neighborhoods such as Concord, Grant City, and New Dorp Beach have prices that are relatively close to each other, ranging between dollar 57 and dollar 58. This suggests that some neighborhoods in NYC, particularly in Staten Island and the Bronx, offer more affordable Airbnb listings, making them attractive to budget-conscious travelers.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, these insights can definitely have a positive business impact:

- Targeted Marketing: Airbnb hosts in these neighborhoods can emphasize affordability in their listings, attracting budget-conscious travelers.
- Strategic Pricing: Hosts in these neighborhoods can adjust their prices to remain competitive and increase bookings.
- Airbnb Growth: Understanding the price sensitivity of certain neighborhoods can help Airbnb design specific campaigns, offering promotions or highlighting these neighborhoods in their recommendations to travelers.

By focusing on these neighborhoods, Airbnb can attract more customers who are looking for affordable stays in NYC, increasing overall platform usage and revenue.

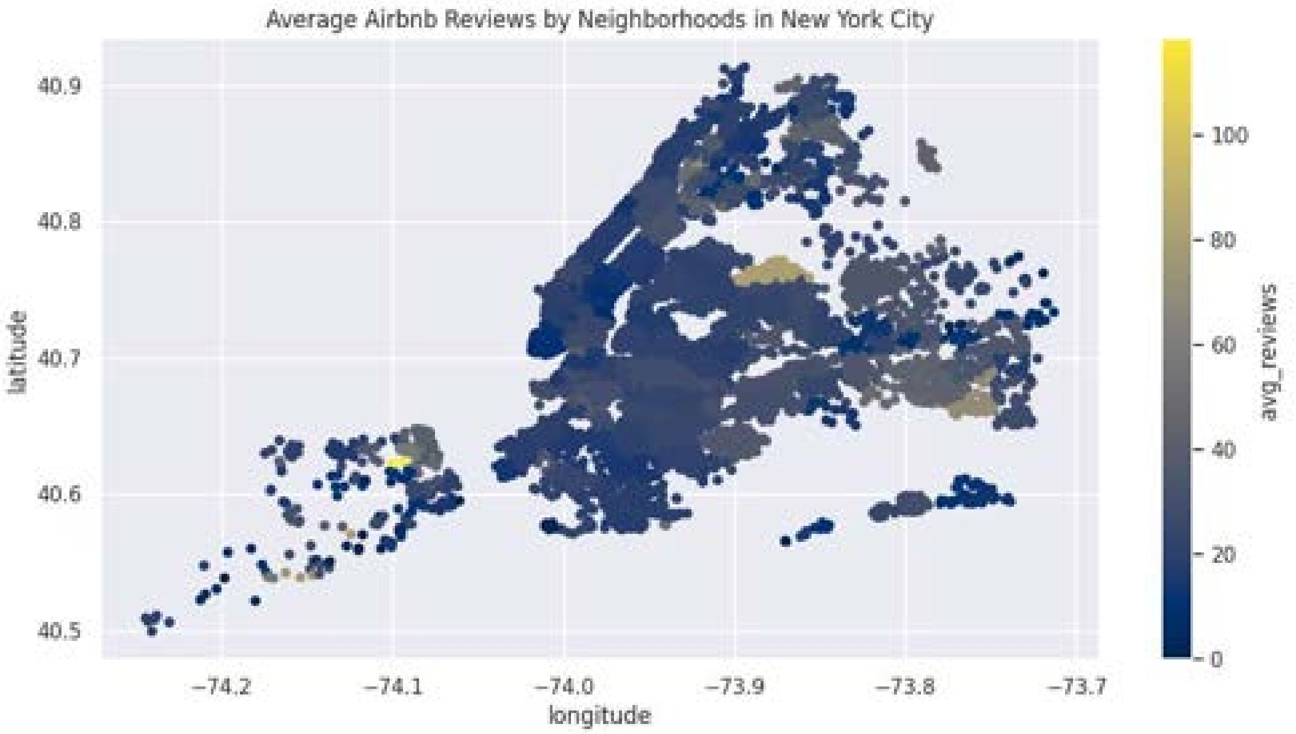
Yes, there could be potential negative growth due to price undercutting. If hosts in these least expensive neighborhoods attempt to lower their prices further in a bid to attract more customers, it could lead to:

- Decreased Profitability: Hosts may struggle to cover operational costs, maintenance, or even make a profit if prices drop too low.
- Quality Compromise: Lower prices may result in a reduction in the quality of service, which could lead to negative reviews, driving customers away in the long term.

For Airbnb, a significant shift towards lower-priced listings may also reduce the perception of quality on the platform, negatively affecting brand image.

Chart - 17 : Scatter Plot : Distribution of Average Airbnb Reviews Across New York City Neighborhoods

```
# Group the data by neighborhood and calculate the average number of reviews
neighborhood_avg_reviews = Airbnb_df.groupby('neighbourhood')['total_reviews'].mean()

# Create a new DataFrame with the average number of reviews for each neighborhood
neighbourhood_reviews = pd.DataFrame({"neighbourhood": neighborhood_avg_reviews.index, "a

# Merge the average number of reviews data with the original DataFrame
df = Airbnb_df.merge(neighbourhood_reviews, on="neighbourhood")

# Create the scattermapbox plot
fig = df.plot.scatter(x="longitude", y="latitude", c="avg_reviews", title="Average Airbnb

# Display the scatter map
plt.show()
```



1. Why did you pick the specific chart?

The scattermap of New York City's neighborhoods plotted against longitude and latitude, colored by the average number of reviews, was chosen because it provides a clear geographical visualization of how Airbnb review activity is distributed across the city. By mapping the review intensity across neighborhoods, it becomes easier to spot regional patterns in guest feedback.

2. What is/are the insight(s) found from the chart?

- High Review Density in Popular Areas: Certain neighborhoods, particularly in Manhattan and parts of Brooklyn, exhibit higher concentrations of average reviews. This could indicate areas that are popular among travelers, offering a high volume of Airbnb properties that attract more guests.
- Outlying Neighborhoods: Lower review activity is apparent in peripheral neighborhoods, which might reflect fewer Airbnb listings or less tourist interest.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- Targeting High-Review Areas for Marketing: Areas with high average reviews, such as parts of Manhattan, could be further promoted in marketing campaigns to boost the attractiveness of these neighborhoods. Hosts in these areas might be encouraged to maintain high service quality to continue receiving positive feedback.
- Investment in Popular Locations: For hosts or potential investors, understanding which neighborhoods consistently receive higher reviews could inform strategic decisions on where to purchase or rent properties to maximize returns.

Need to review:

- Low Review Areas: Certain neighborhoods with consistently lower reviews may signal areas with less tourist demand or potential service issues. Hosts in these regions might experience difficulty in maintaining high occupancy rates or achieving competitive pricing, leading to possible negative growth unless they adapt their strategies.
- Service Gaps in High-Volume Areas: If the high-review neighborhoods experience service issues or are overwhelmed by demand, it could lead to a decrease in guest satisfaction and, in the long run, reduced bookings.

Multivariate Analysis (3 or more variables)

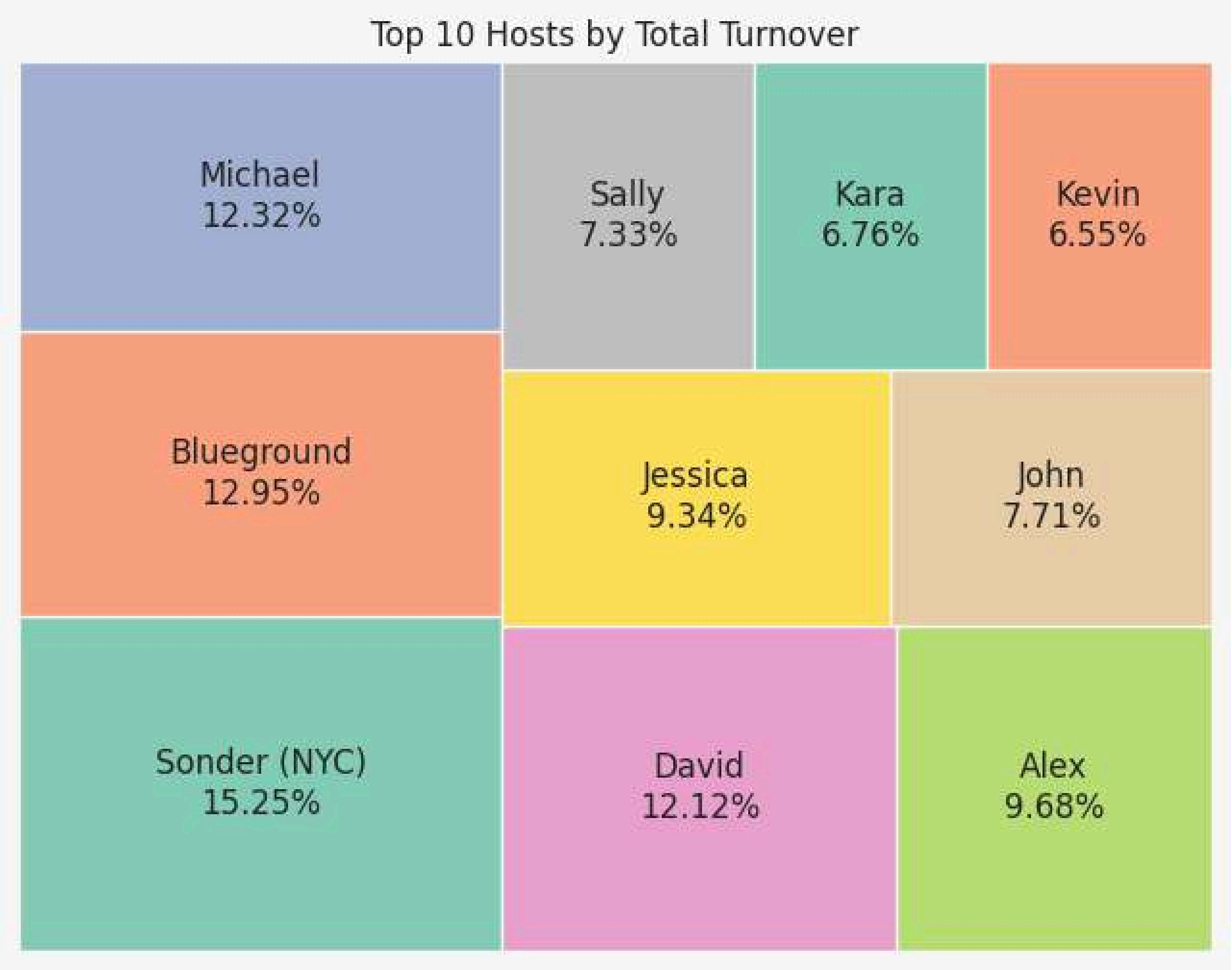
Chart - 18 : Tree Map : Top 10 Hosts in Total Turnover

```
# Calculate turnover for each host
host_turnover = Airbnb_df.groupby('host_name')['price'].sum().reset_index(name='total_tur

# Sort hosts by turnover in descending order
top_hosts = host_turnover.sort_values(by='total_turnover', ascending=False).head(10)

# Extract host names and total turnover into two separate lists
host_names = top_hosts['host_name'].tolist()
total_turnovers = top_hosts['total_turnover'].tolist()

# Print the lists to verify
print("Host Names:", host_names)
print("Total Turnovers:", total_turnovers)
```

```
Host Names: ['Sonder (NYC)', 'Blueground', 'Michael', 'David', 'Alex', 'Jessica', 'Jc
Total Turnovers: [82795, 70331, 66895, 65844, 52563, 50697, 41892, 39789, 36723, 3555
```

```
!pip install squarify

import squarify

# Prepare the data for the treemap
sizes = total_turnovers
labels = host_names
colors = plt.get_cmap('Set2').colors

# Calculate percentages
total = sum(sizes)
percentages = [f'{(size / total) * 100:.2f}%' for size in sizes]

# Combine host names with percentages
full_labels = [f'{host}\n{perc}' for host, perc in zip(labels, percentages)]

plt.figure(figsize=(8, 6), facecolor='#F7F7F7')
squarify.plot(sizes=sizes, label=full_labels, color=colors, alpha=0.8)
plt.title('Top 10 Hosts by Total Turnover')
plt.axis('off')  # Hide the axis
plt.show()
```

```
Collecting squarify
  Downloading squarify-0.4.4-py3-none-any.whl.metadata (600 bytes)
Downloading squarify-0.4.4-py3-none-any.whl (4.1 kB)
Installing collected packages: squarify
Successfully installed squarify-0.4.4
```



1. Why did you pick the specific chart?

I picked the treemap chart because it effectively displays hierarchical data in a compact form, allowing for easy comparison of different hosts' contributions to total turnover. The size of each rectangle visually represents the proportion of turnover, making it simple to identify top performers at a glance.

2. What is/are the insight(s) found from the chart?

The key insights from the chart are:

- Sonder (NYC) is the top performer with 15.25% of total turnover.
- The top three hosts (Sonder, Blueground, and Michael/David) account for over 40% of the total turnover.
- There's a significant gap between the top performers and the bottom performers (e.g., Sonder at 15.25% vs. Kevin at 6.55%).

- The distribution of turnover is relatively uneven, with a few hosts dominating the upper percentages.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

These insights can help create positive business impact by:

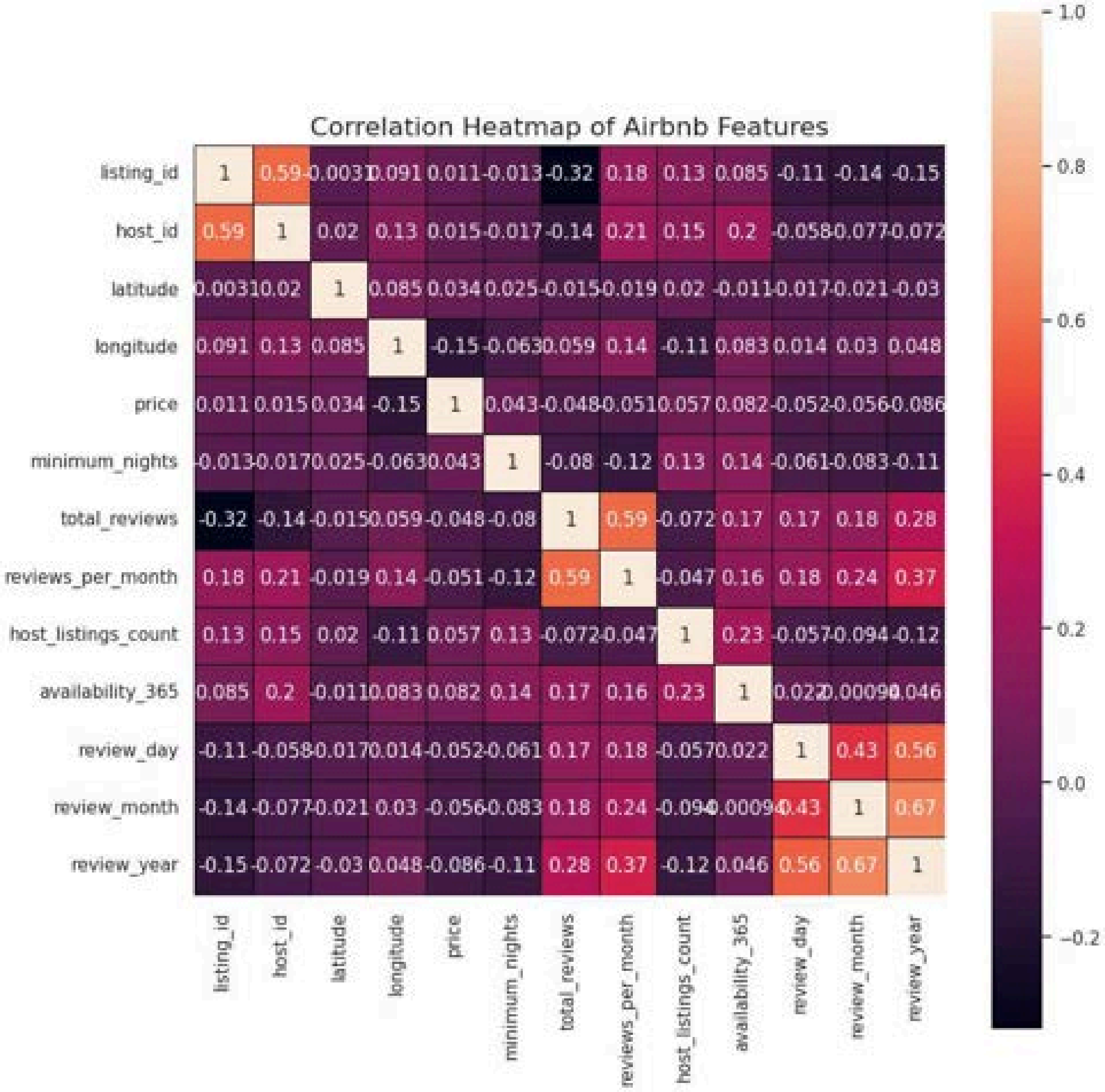
- Identifying top-performing hosts for potential replication of their strategies.
- Highlighting opportunities for improvement among lower-performing hosts.
- Informing resource allocation and support decisions based on host performance.
- Providing a benchmark for setting performance goals for hosts.

While there are no explicit insights leading to negative growth, the chart reveals potential areas of concern:

- The significant disparity between top and bottom performers could indicate inconsistency in host quality or management, which might need addressing.
- Over-reliance on a few top performers (like Sonder NYC) could be a risk if their performance were to decline.
- Lower-performing hosts (e.g., Kevin, Kara) might need additional support or review to improve their turnover and prevent potential negative impact on overall business growth.

Chart - 19 : Heatmap : Airbnb Feature Correlation Analysis

```
# Compute the correlation matrix for numerical columns only
corrmat = Airbnb_df.corr(numeric_only=True)

# Create a figure and axis with a specific size (10x10)
f, ax = plt.subplots(figsize=(10, 10))

# # Visualize correlations as a heatmap
sns.heatmap(corrmat, annot=True, linewidths=0.5, linecolor='black', square=True)

# Add a title to the heatmap for context
plt.title('Correlation Heatmap of Airbnb Features', fontsize=16)

# Adjust the layout to ensure all elements fit nicely within the figure
plt.tight_layout()

# Display the heatmap
plt.show()
```



1. Why did you pick the specific chart?

The heatmap was chosen because it is one of the most effective ways to visually represent the correlation between numerical variables in a dataset. It allows us to quickly identify both positive and negative correlations between the features of the Airbnb dataset. The color gradient intuitively shows the strength and direction of the relationships between variables, making it easy to spot patterns and trends.

2. What is/are the insight(s) found from the chart?

From the heatmap:

- Host ID and Listing ID have a strong positive correlation (0.58), which suggests that there might be hosts with multiple listings.
- Total reviews and reviews per month are strongly correlated (0.59), which makes sense since more reviews per month should translate into a higher total review count.
- Review year, month, and day show significant correlation with each other, particularly between review year and review month (0.66), suggesting a temporal relationship in how reviews are recorded.
- Total reviews and price have a weak negative correlation (-0.31), suggesting that higher-priced listings may receive fewer reviews, likely due to limited affordability.
- Host listings count and availability 365 have a weak positive correlation (0.23), implying