# Alessandro Benevelli Capstone Project IBM Analyze Open Data Sets with Python

# Capstone Project Creation

## IBM SkillsBuild Europe Delivery - Data Analytics

# Pre-requisite

- Understanding of Python, Power BI or Tableau
- Understanding of Data Cleaning
- Understanding Data Visualization

### Level of Exercise: Intermediate

### Duration: approximately 3 hours

## Data Analytics of Airbnb Data:

### Objective:

In this exericise, you will be performing Data Analytics on an Open Dataset dataset coming from Airbnb. Some of the tasks include

- Data Cleaning.
- Data Transformation
- Data Visualization.

### Overview of Airbnb Data:

People's main criteria when visiting new places are reasonable accommodation and food. Airbnb (Air-Bed-Breakfast) is an online marketplace created to meet this need of people by renting out their homes for a short term. They offer this facility at a relatively lower price than hotels. Further people worldwide prefer the homely and economical service offered by them. They offer services across various geographical locations

### Dataset Source

YOu can get the dataset for this assessment using the following link:
https://www.kaggle.com/datasets/arianazmoudeh/airbnbopendata

This dataset contains information such as the neighborhood offering these services, room type, price,avaliabilty, reviews, service fee, cancellation policy and rules to use the house. This analysis will help airbnb in improving its services.

So all the best for your Data Analytics Journey on Airbnb data!!!

# Task 1: Data Loading (Python)

1. Read the csv file and load it into a pandas dataframe.
2. Display the first five rows of your dataframe.
3. Display the data types of the columns.

In [66]:
```
##Import Libraries

import pandas as pd
import numpy as np

# visualization
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

In [ ]:
```
## Read the csv file

df = pd.read_csv(r"C:\Users\aless\OneDrive\Documenti\Final_Project_IBM\Airbnb_Open_
```

In [68]:
```
## Display the first 5 rows
df.head()
```

Out[68]:

| | id | NAME | host id | host_identity_verified | host name | neighbourhood group | neighbo |
|---|---|---|---|---|---|---|---|
| **0** | 1001254 | Clean & quiet apt home by the park | 80014485718 | unconfirmed | Madaline | Brooklyn | Ke |
| **1** | 1002102 | Skylit Midtown Castle | 52335172823 | verified | Jenna | Manhattan | |
| **2** | 1002403 | THE VILLAGE OF HARLEM....NEW YORK ! | 78829239556 | NaN | Elise | Manhattan | |
| **3** | 1002755 | NaN | 85098326012 | unconfirmed | Garry | Brooklyn | Cli |
| **4** | 1003689 | Entire Apt: Spacious Studio/Loft by central park | 92037596077 | verified | Lyndon | Manhattan | Eas |

5 rows × 26 columns

In [69]:
```
## Display the data types
df.dtypes
```

Out[69]:
```
id                                int64
NAME                             object
host id                           int64
host_identity_verified           object
host name                        object
neighbourhood group              object
neighbourhood                    object
lat                             float64
long                            float64
country                          object
country code                     object
instant_bookable                 object
cancellation_policy              object
room type                        object
Construction year               float64
price                            object
service fee                      object
minimum nights                  float64
number of reviews               float64
last review                      object
reviews per month               float64
review rate number              float64
calculated host listings count  float64
availability 365                float64
house_rules                      object
license                          object
dtype: object
```

# Task 2a: Data Cleaning (Any Tool)

1. Drop some of the unwanted columns. These include `host id`, `id`, `country` and `country code` from the dataset.
2. State the reason for not including these columns for your Data Analytics.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots before and after the elimination of the columns.

In [70]:
```
unwanted_columns = ['host id', 'id', 'country', 'country code']
df_cleaned = df.drop(columns=unwanted_columns)
```

In [71]:
```
print(df_cleaned.head())
```

```
                                                NAME host_identity_verified  \
0                Clean & quiet apt home by the park            unconfirmed
1                             Skylit Midtown Castle               verified
2               THE VILLAGE OF HARLEM....NEW YORK !                    NaN
3                                               NaN            unconfirmed
4  Entire Apt: Spacious Studio/Loft by central park               verified

  host name neighbourhood group neighbourhood       lat      long  \
0  Madaline            Brooklyn    Kensington  40.64749 -73.97237
1     Jenna           Manhattan       Midtown  40.75362 -73.98377
2     Elise           Manhattan        Harlem  40.80902 -73.94190
3     Garry            Brooklyn  Clinton Hill  40.68514 -73.95976
4    Lyndon           Manhattan   East Harlem  40.79851 -73.94399

  instant_bookable cancellation_policy        room type  ...  service fee  \
0            False              strict     Private room  ...         $193
1            False            moderate  Entire home/apt  ...          $28
2             True            flexible     Private room  ...         $124
3             True            moderate  Entire home/apt  ...          $74
4            False            moderate  Entire home/apt  ...          $41

  minimum nights number of reviews  last review  reviews per month  \
0           10.0               9.0   10/19/2021               0.21
1           30.0              45.0    5/21/2022               0.38
2            3.0               0.0          NaN                NaN
3           30.0             270.0    7/5/2019               4.64
4           10.0               9.0   11/19/2018               0.10

  review rate number  calculated host listings count  availability 365  \
0                4.0                             6.0             286.0
1                4.0                             2.0             228.0
2                5.0                             1.0             352.0
3                4.0                             1.0             322.0
4                3.0                             1.0             289.0

                                        house_rules  license
0  Clean up and treat the home the way you'd like...      NaN
1  Pet friendly but please confirm with me if the...      NaN
2  I encourage you to use my kitchen, cooking and...      NaN
3                                               NaN      NaN
4  Please no smoking in the house, porch or on th...      NaN

[5 rows x 22 columns]
```

# Task 2b: Data Cleaning (Python)

- Check for missing values in the dataframe and display the count in ascending order. **If the values are missing, impute the values as per the datatype of the columns.**
- Check whether there are any duplicate values in the dataframe and, if present, remove them.
- Display the total number of records in the dataframe before and after removing the duplicates.

In [72]:
```
## Check for missing values in the dataframe and display the count in ascending ord
missing_data = df.isnull().sum().sort_values(ascending=True)
print("Missing Values:")
print(missing_data)
```

```
Missing Values:
id                                    0
room type                             0
host id                               0
long                                  8
lat                                   8
neighbourhood                        16
neighbourhood group                  29
cancellation_policy                  76
instant_bookable                    105
country code                        131
number of reviews                   183
Construction year                   214
price                               247
NAME                                250
service fee                         273
host_identity_verified              289
calculated host listings count      319
review rate number                  326
host name                           406
minimum nights                      409
availability 365                    448
country                             532
reviews per month                 15879
last review                       15893
house_rules                       52131
license                          102597
dtype: int64
```

In [73]:
```
## Check whether there are any duplicate values in the dataframe and if present rem
df.duplicated().sum()
```

Out[73]:
```
541
```

In [74]:
```
## Display the total number of records in the dataframe after removing the duplicat
# Impute missing values based on datatype of columns
for column, dtype in df.dtypes.items():
    if missing_data[column] > 0:
        if dtype == 'object':
            # If the column is of object type, impute missing values with the mode
            df[column].fillna(df[column].mode()[0], inplace=True)
        else:
            # For numeric columns, impute missing values with the mean
            df[column].fillna(df[column].mean(), inplace=True)
```

In [75]:
```
# Check for and remove duplicates
before_duplicates = df.shape[0]  # Total number of records before removing duplicat
df.drop_duplicates(inplace=True)
after_duplicates = df.shape[0]  # Total number of records after removing duplicates

# Display the total number of records before and after removing duplicates
print(f"Total number of records before removing duplicates: {before_duplicates}")
print(f"Total number of records after removing duplicates: {after_duplicates}")
```

```
Total number of records before removing duplicates: 102599
Total number of records after removing duplicates: 102058
```

# Task 3: Data Transformation (Any Tool)

- Rename the column `availability 365` to `days_booked`

- Convert all column names to lowercase and replace the spaces in the column names with an underscore "_".
- Remove the dollar sign and comma from the columns `price` and `service_fee`. If necessary, convert these two columns to the appropriate data type.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots of your work.

In [76]:
```
## Rename the column.

df.rename(columns={'availability_365': 'days_booked'}, inplace=True)
```

In [77]:
```
## Convert all column names to lowercase and replace the spaces with an underscore
df.columns = df.columns.str.lower().str.replace(' ', '_')
```

In [78]:
```
## Remove the dollar sign and comma from the columns. If necessary, convert these t
columns_to_clean = ['price', 'service_fee']

# Remove '$' and ',' and convert to appropriate data type
for column in columns_to_clean:
    df[column] = df[column].replace({'\$': '', ',': ''}, regex=True).astype(float)

# Display the transformed DataFrame
print(df.head())
```

```
        id                                              name      host_id  \
0  1001254                Clean & quiet apt home by the park  80014485718
1  1002102                             Skylit Midtown Castle  52335172823
2  1002403               THE VILLAGE OF HARLEM....NEW YORK !  78829239556
3  1002755                                 Home away from home  85098326012
4  1003689  Entire Apt: Spacious Studio/Loft by central park  92037596077

  host_identity_verified host_name neighbourhood_group neighbourhood  \
0            unconfirmed  Madaline            Brooklyn    Kensington
1               verified     Jenna           Manhattan       Midtown
2            unconfirmed     Elise           Manhattan        Harlem
3            unconfirmed     Garry            Brooklyn  Clinton Hill
4               verified    Lyndon           Manhattan   East Harlem

        lat      long        country  ... service_fee  minimum_nights  \
0  40.64749 -73.97237  United States  ...       193.0            10.0
1  40.75362 -73.98377  United States  ...        28.0            30.0
2  40.80902 -73.94190  United States  ...       124.0             3.0
3  40.68514 -73.95976  United States  ...        74.0            30.0
4  40.79851 -73.94399  United States  ...        41.0            10.0

  number_of_reviews last_review  reviews_per_month  review_rate_number  \
0               9.0  10/19/2021           0.210000                 4.0
1              45.0   5/21/2022           0.380000                 4.0
2               0.0   6/23/2019           1.374022                 5.0
3             270.0    7/5/2019           4.640000                 4.0
4               9.0  11/19/2018           0.100000                 3.0

   calculated_host_listings_count  availability_365  \
0                             6.0             286.0
1                             2.0             228.0
2                             1.0             352.0
3                             1.0             322.0
4                             1.0             289.0

                                        house_rules   license
0  Clean up and treat the home the way you'd like...  41662/AL
1  Pet friendly but please confirm with me if the...  41662/AL
2  I encourage you to use my kitchen, cooking and...  41662/AL
3                                            #NAME?  41662/AL
4  Please no smoking in the house, porch or on th...  41662/AL

[5 rows x 26 columns]
```

## Task 4: Exploratory Data Analysis (Any Tool)

- List the count of various room types avaliable in the dataset.
- Which room type has the most strict cancellation policy?
- List the average price per neighborhood group, and highlight the most expensive neighborhood to rent from.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots of your work.

In [79]:
```
## List the count of various room types avaliable with Airbnb

room_type_count = df['room_type'].value_counts()

# Room type with the most strict cancellation policy
most_strict_cancel = df[df['cancellation_policy'] == df['cancellation_policy'].max(
```

```
print("Count of various room types available:")
print(room_type_count)
print("\nRoom type with the most strict cancellation policy:", most_strict_cancel)
```

```
Count of various room types available:
room_type
Entire home/apt    53429
Private room       46306
Shared room         2208
Hotel room           115
Name: count, dtype: int64

Room type with the most strict cancellation policy: Private room
```

In [80]:
```
# Average price per neighborhood group
avg_price_neighborhood = df.groupby('neighbourhood_group')['price'].mean().sort_val
most_expensive_neighborhood = avg_price_neighborhood.idxmax()
most_expensive_price = avg_price_neighborhood.max()


print("\nAverage price per neighborhood group:")
print(avg_price_neighborhood)
print("\nMost expensive neighborhood to rent from:", most_expensive_neighborhood, f
```

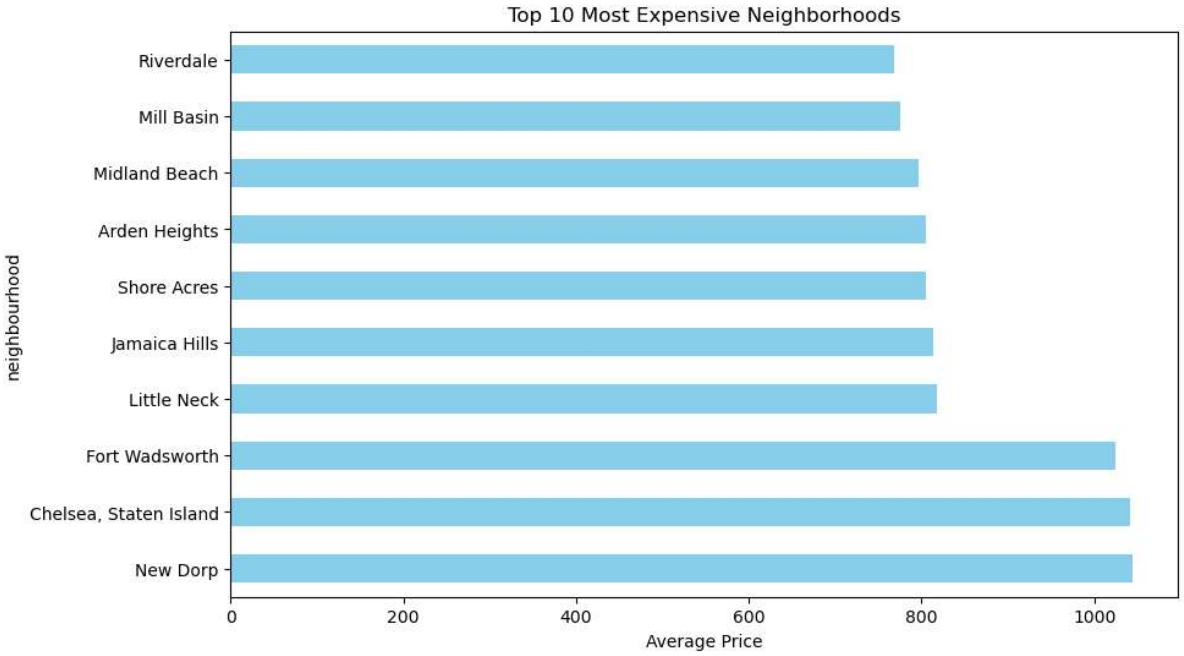
```
Average price per neighborhood group:
neighbourhood_group
Queens           629.156248
Bronx            626.668894
Brooklyn         625.562575
Staten Island    622.669125
Manhattan        621.612789
brookln          580.000000
manhatan         460.000000
Name: price, dtype: float64

Most expensive neighborhood to rent from: Queens ($629.16)
```

# Task 5a: Data Visualization (Any Tool)

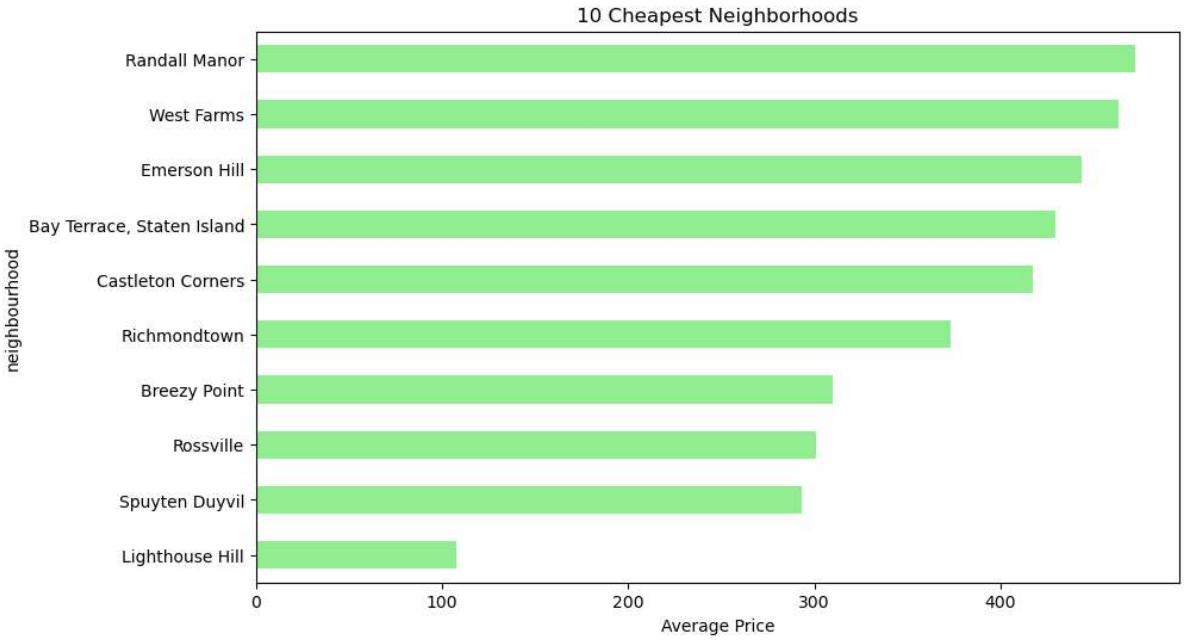
- Create a horizontal bar chart to display the top 10 most expensive neighborhoods in the dataset.
  - Create another chart with the 10 cheapest neighborhoods in the dataset.
- Create a box and whisker chart that showcases the price distribution of all listings split by room type.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots of your work.

In [81]:
```
top_expensive_neighborhoods = df.groupby('neighbourhood')['price'].mean().nlargest(
plt.figure(figsize=(10, 6))
top_expensive_neighborhoods.plot(kind='barh', color='skyblue')
plt.xlabel('Average Price')
plt.title('Top 10 Most Expensive Neighborhoods')
plt.show()
```




In [82]:
```
cheapest_neighborhoods = df.groupby('neighbourhood')['price'].mean().nsmallest(10)
plt.figure(figsize=(10, 6))
cheapest_neighborhoods.plot(kind='barh', color='lightgreen')
plt.xlabel('Average Price')
plt.title('10 Cheapest Neighborhoods')
plt.show()
```




In [83]:
```
plt.figure(figsize=(10, 6))
sns.boxplot(x='room_type', y='price', data=df, palette='pastel')
plt.xlabel('Room Type')
plt.ylabel('Price')
plt.title('Price Distribution by Room Type')
plt.show()
```

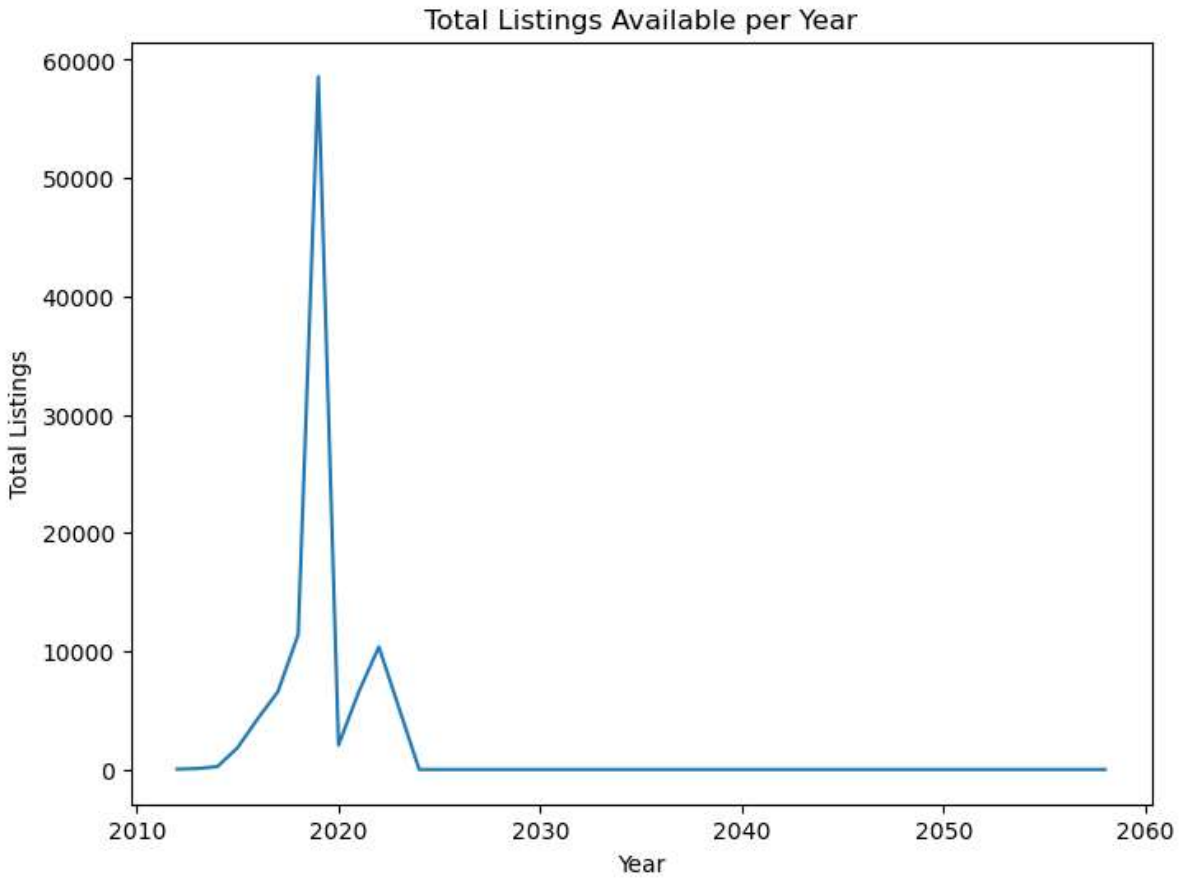

# Task 5b: Data Visualization (Any Tool)

- Create a scatter plot to illustrate the relationshi between the service fee and the room price and write down the kind of correlation, if any, that you see.
- Create a line chart to showcase the total amount of listings available per year.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots of your work.

In [86]:
```
import seaborn as sns
plt.figure(figsize=(8, 6))
sns.scatterplot(data=df, x='service_fee', y='price')
plt.xlabel('service_fee')
plt.ylabel('Room Price')
plt.title('Relationship between Service fee and Room Price')
plt.show()
```


In [88]:
```
df['year'] = pd.to_datetime(df['last_review']).dt.year

plt.figure(figsize=(8, 6))
listings_per_year = df['year'].value_counts().sort_index()
sns.lineplot(x=listings_per_year.index, y=listings_per_year.values)
plt.xlabel('Year')
plt.ylabel('Total Listings')
plt.title('Total Listings Available per Year')
plt.show()
```



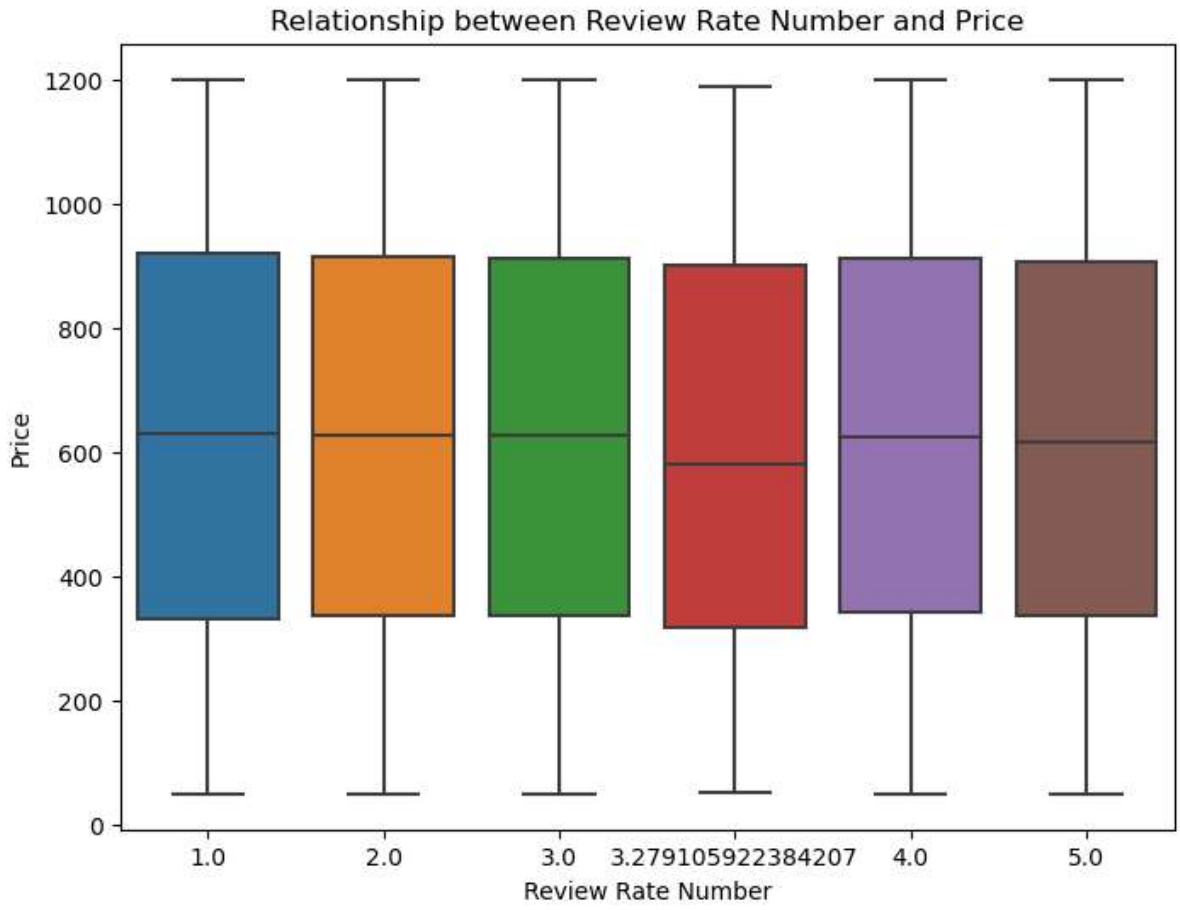

# Task 5c: Data Visualization (Any Tool)

- Create a data visualization of your choosing using one of the review columns in isolation or in combination with another column.

If using Python for this exercise, please include the code in the cells below. If using any other tool, please include screenshoots of your work.

In [106…

```
# Visualization of 'Review_rate_number' and 'Price'
plt.figure(figsize=(8, 6))
sns.boxplot(data=df, x='review_rate_number', y='price')
plt.xlabel('Review Rate Number')
plt.ylabel('Price')
plt.title('Relationship between Review Rate Number and Price')
plt.show()
```




https://www.credly.com/badges/94cd09c5-fbca-466d-b6a5-b38ac423fbeb/linked_in?t=s3wbqe