**Airbnb NYC SQL Analysis Project**

**Exploring Airbnb Listings Using Structured Query Language(SQL)**

**Author: Mohd Farhan Khan**

**Short Description**:

This report presents an analysis of the Airbnb NYC 2019 dataset using SQL queries, highlighting key business insights and data-driven recommendations.

**Table of Contents**

1. Project Overview
2. Skills Used
3. Data Cleaning & Preparation
4. SQL Business Questions and Insights
5. Conclusion
6. **Project Overview**

This project focuses on analyzing the Airbnb NYC 2019 dataset using SQL. The main objectives were to clean the raw dataset, create helpful new features, and answer 15 real-world business questions related to pricing, availability, host behavior, and neighborhood trends.

I applied key SQL techniques such as data aggregation, joins, window functions (RANK), and Common Table Expressions (CTEs) to derive meaningful insights from the data. Through this project, I gained hands-on experience with handling real-world datasets and strengthened my skills in data cleaning, structured querying, and insight generation.

1. **Skills Used**

 **SQL Basics:** SELECT, WHERE, GROUP BY, HAVING, ORDER BY, LIMIT

 **Joins:** INNER JOIN to combine data from multiple tables

 **Aggregation Functions:** COUNT, AVG, SUM, FLOOR

 **Window Functions:** RANK () to rank hosts within neighborhood groups

 **CTEs (Common Table Expressions):** To structure complex queries simply

 **Data Cleaning:** Handling missing values, correcting data types, standardizing text columns

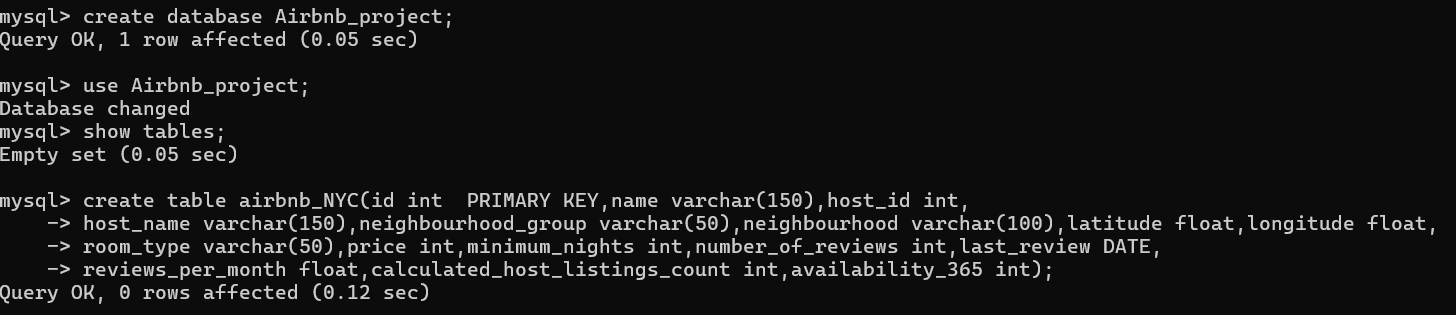
 **Feature Engineering:** Creating new columns (price category, review month, host popularity)

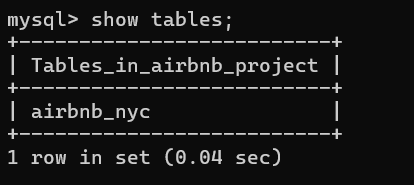
 **Analytical Thinking:** Writing insights based on SQL query outputs

 **Problem Solving:** Handling large datasets and real-world messy data

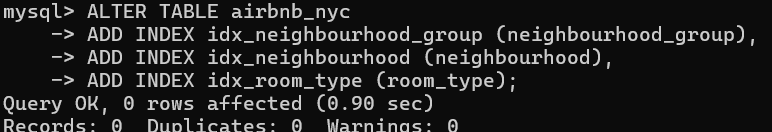
**3. Data Cleaning & Preparation**

Q) creating the database and table:

.

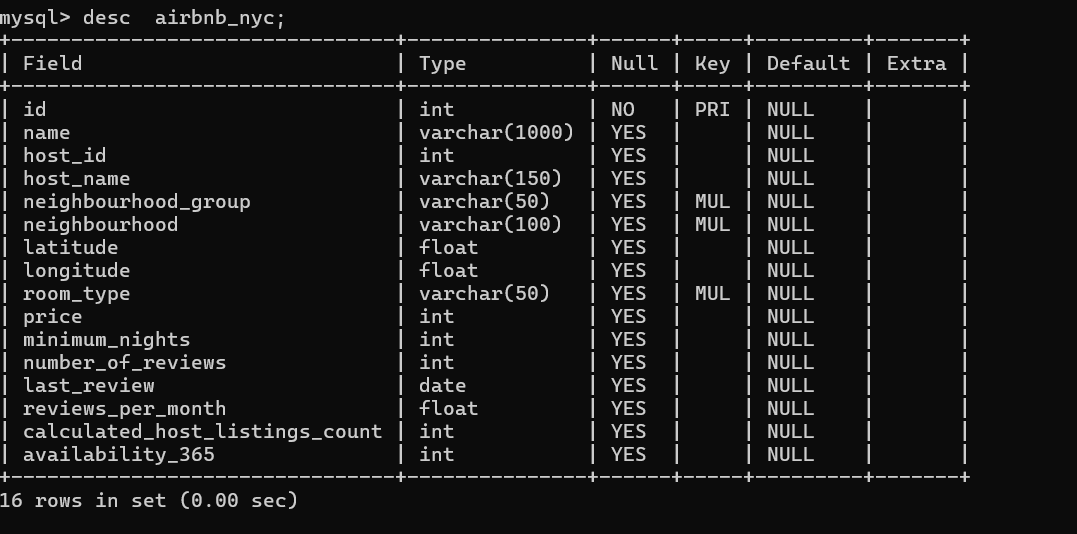


**Q) Adding indexes** for faster queries:



Q) showing the structure of the table airbnb\_nyc and

checking the Data type ?



Insights: Checked the Data Types with (describe) command and Our Data Types are Fine.

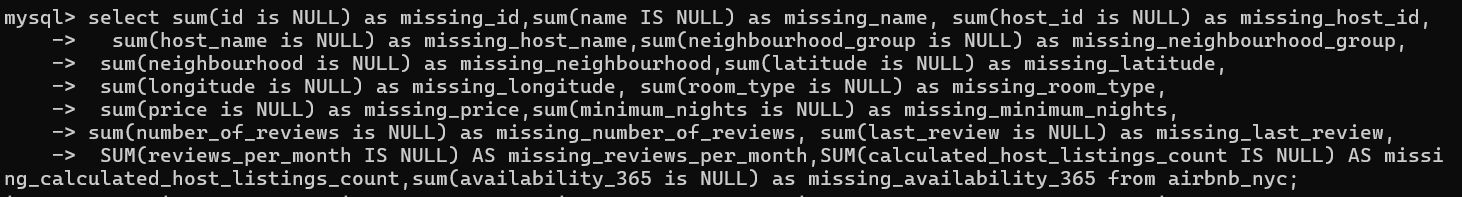
Q) Removing The Duplicates:



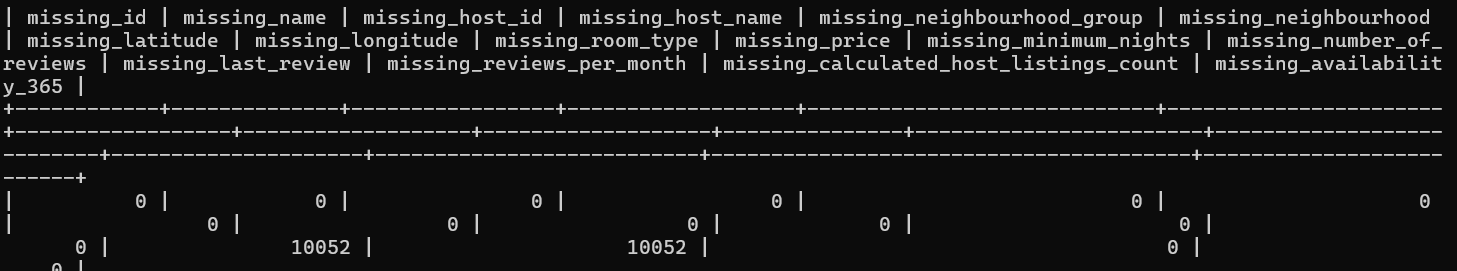
Insights:

There are no duplicate ids in our table airbnb\_nyc. Our Data is already Clean.

Q) Finding out missing (Null) values from all columns at once:



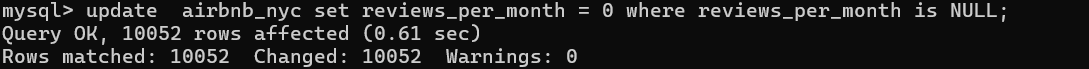
Output



Insights:

Only Two columns last\_review and reviews\_per\_month have 10,052 missing values.

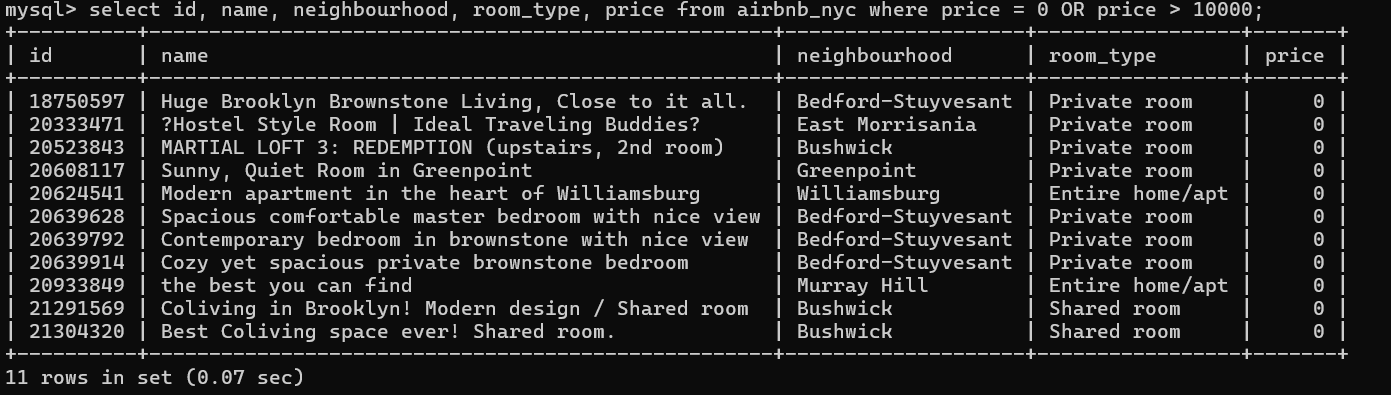
Q) Handling missing reviews\_per\_month and last\_review:



Insights:

Replaced missing reviews\_per\_month with 0 and leaving last\_review as NULL because it means No review\_ which is True information.

Q) Removing unrealistic prices and minimum nights:



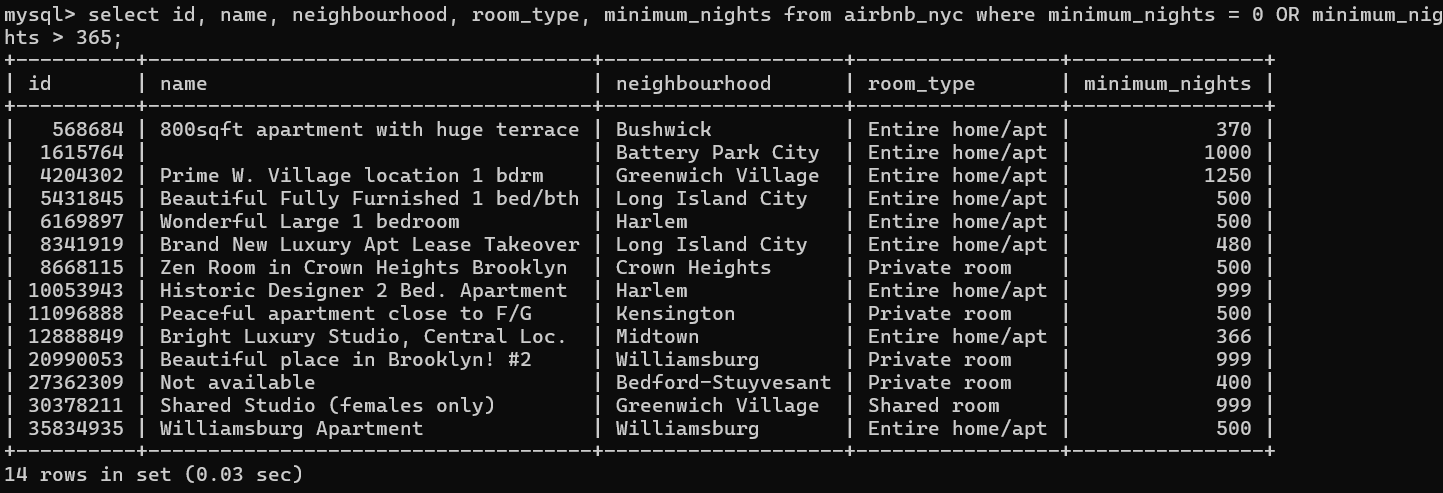
Here, found out some unrealistic prices with the above Query and then deleted them with this Query.



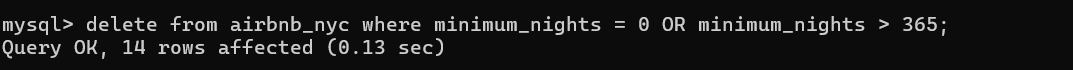
Hence, confirmed with the below Query that there is no more unrealistic price value.



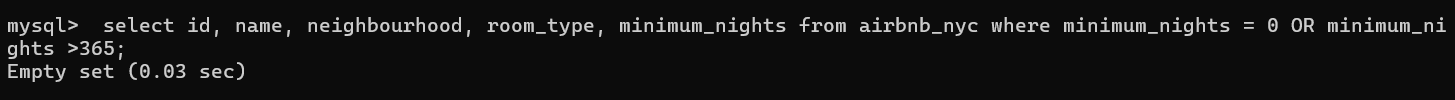
Now, detecting the unrealistic minimum\_nights with this Query



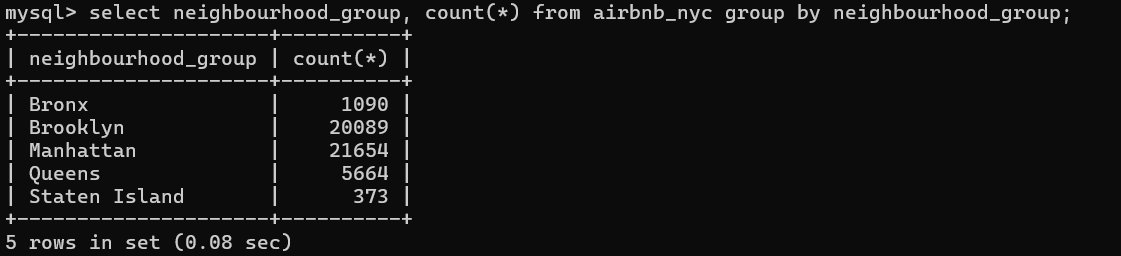
And then deleting them with this Query



Hence, confirmed with the below Query that there is no more unrealistic minimum\_nights value.



**Q) Why Standardizing the Text columns** ?



Because There are Hidden issues in the raw CSV data like :-

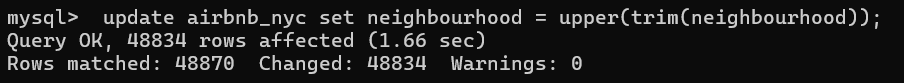
1. **Inconsistent Capitalization**: "Bronx", "Brooklyn", "Manhattan", etc. all are **capitalized correctly** (B, M, Q). But what if some records had: Brooklyn, BROOKLYN, Bronx, BROnx etc.
2. **Hidden Spaces Problem**: If even **one record** has a **space** like ' Brooklyn' (space at beginning) or 'Brooklyn ' (space at end),

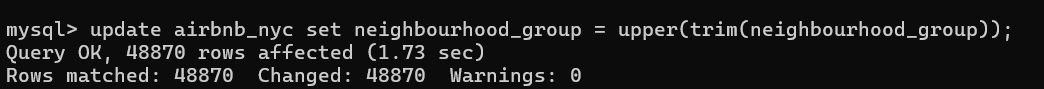
- It **looks same** when you print it on screen

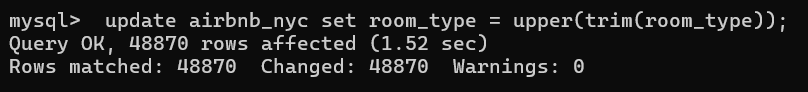
- But Sql treat it as different value.

Hence, To address these issues, I applied data cleaning techniques such as:

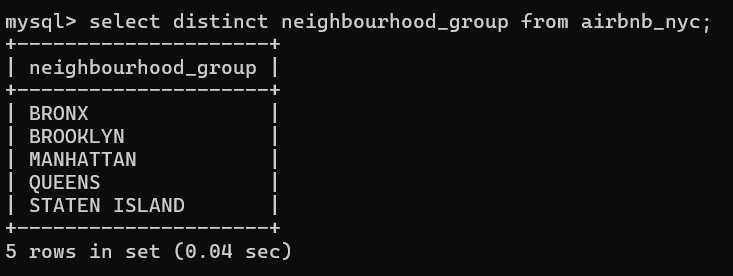
* Standardizing capitalization for consistency.
* Removing leading and trailing spaces using TRIM () or equivalent methods to ensure accurate comparisons in SQL queries."





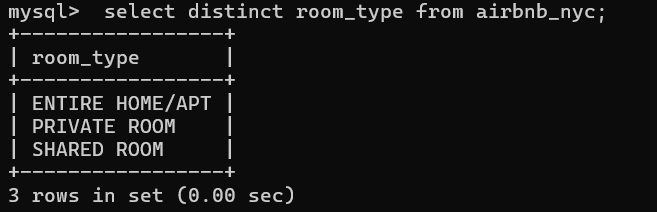


Now, our text columns (neighbourhood\_group, neighbourhood, room\_type) are **fully cleaned and standardized**.







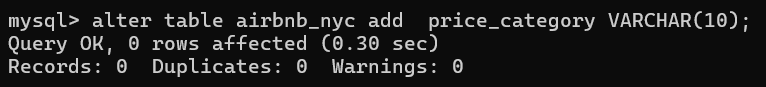


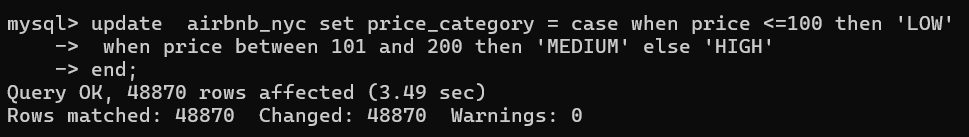
**Q) Why Creating New Helpful Columns** ?

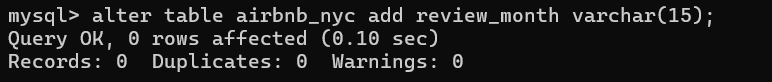
Because creating new columns helps simplify data analysis and makes patterns easier to identify. Hence

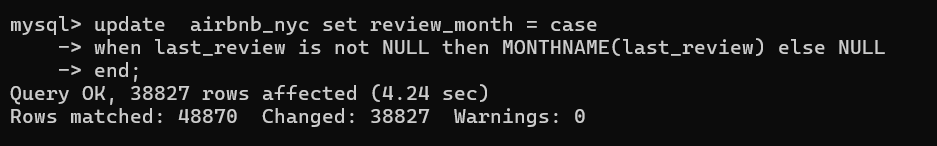
|  |  |
| --- | --- |
| **TASK** | **STATUS** |
| 1) Adding price\_category column (LOW / MEDIUM / HIGH) based on price | Done |
| 2) Adding review\_month column by extracting month name from last\_review | Done |
| 3) Adding host\_popularity column based on number of reviews | Done |

1. Adding price\_category (LOW/MEDIUM/HIGH) for better Comparison and Trend Analysis.

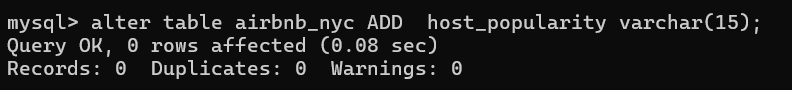


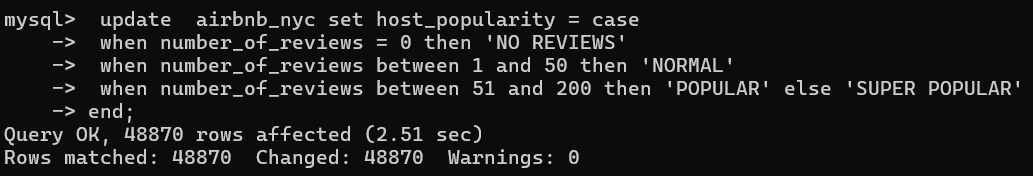


2) Adding review\_month: Extracts the month from last\_review to identify seasonal trends and patterns in reviews. 



3) Creating host\_popularity based on number\_of\_reviews, helping assess host reputation and performance.





**DESCRIPTION:**

In this step, I enhanced the dataset by creating new useful columns:

* **Price Category** to classify listings based on their price.
* **Review Month** to analyze seasonality trends based on reviews.
* **Host Popularity** to categorize hosts based on the number of reviews received.

**Project Data Preparation Summary**

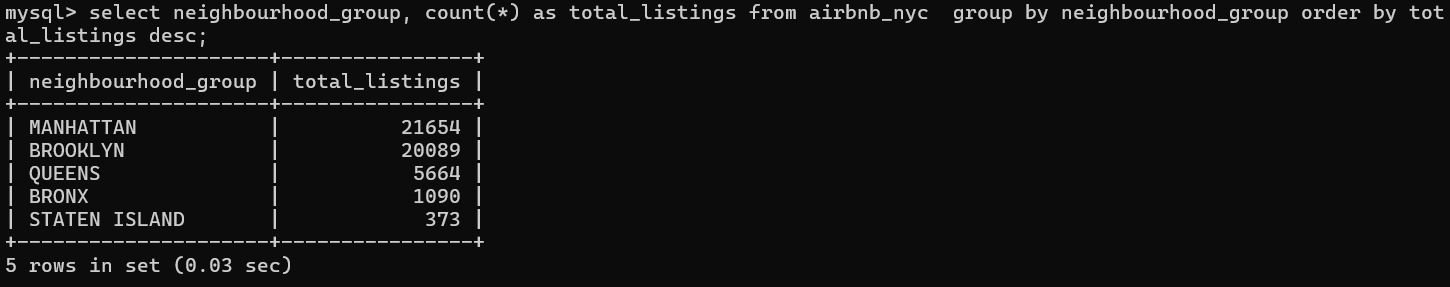
Before starting the SQL analysis, a thorough data preparation process was completed to ensure clean and reliable results. The following steps were successfully performed:

|  |  |
| --- | --- |
| **TASK** | **STATUS** |
| Data Cleaning (duplicates, missing values, type corrections) | **Completed** |
| Table Standardization (text formatting, trimming spaces) | **Completed** |
| Featured Engineering(creation of new columns: price category, review month, host popularity) | **Completed** |
| Ready for SQL Analysis and Insight Generation | **READY** |

Through these steps, the dataset was transformed into a structured and professional format, allowing for accurate analysis and impressive insights.

**4. SQL Questions and Answers Analysis**

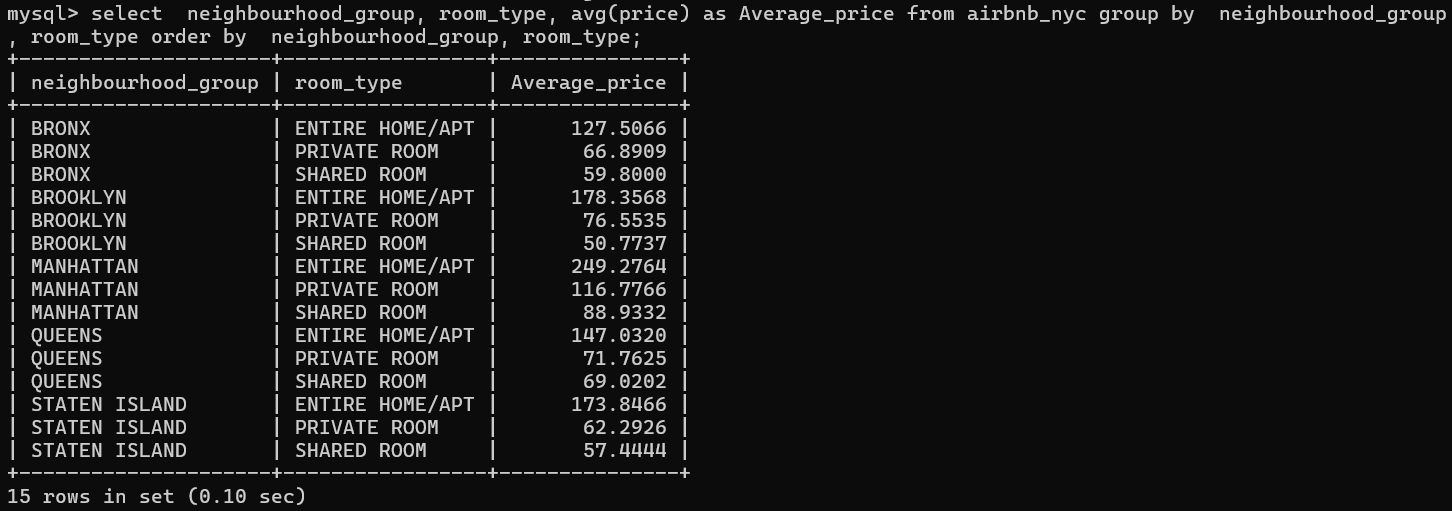
**Q1. What is the total number of listings per neighborhood group?**

****

**Insights:**

Manhattan and Brooklyn dominate the Airbnb market, followed by Queens.

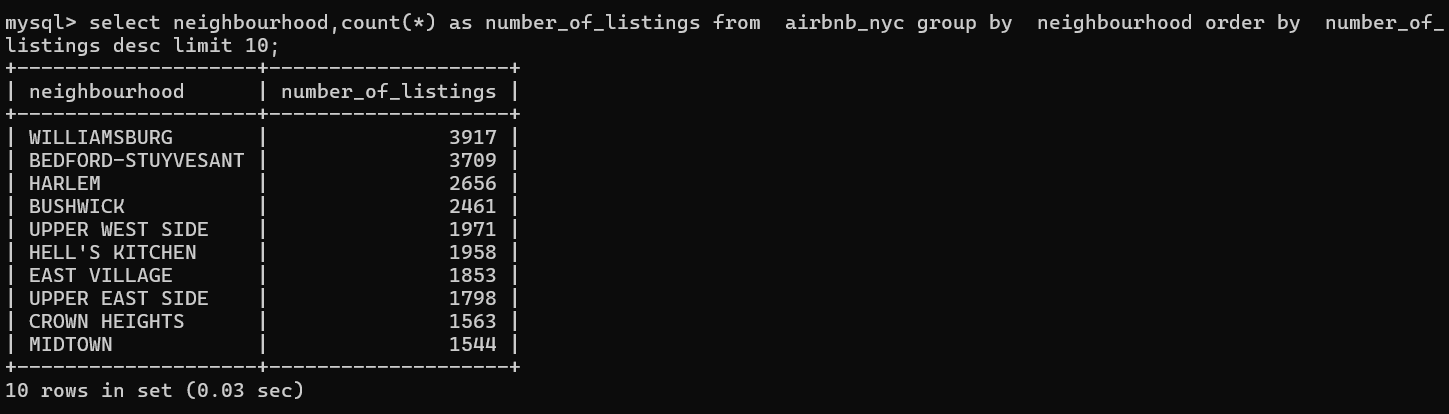
**Q2. What is the average price of listings by room type and neighborhood group?**



**Insights:**

This query shows how the average listing price varies across different neighborhood groups and room types. Typically, Manhattan entire apartments have the highest average prices, while shared rooms in Bronx and Queens are among the most affordable options.

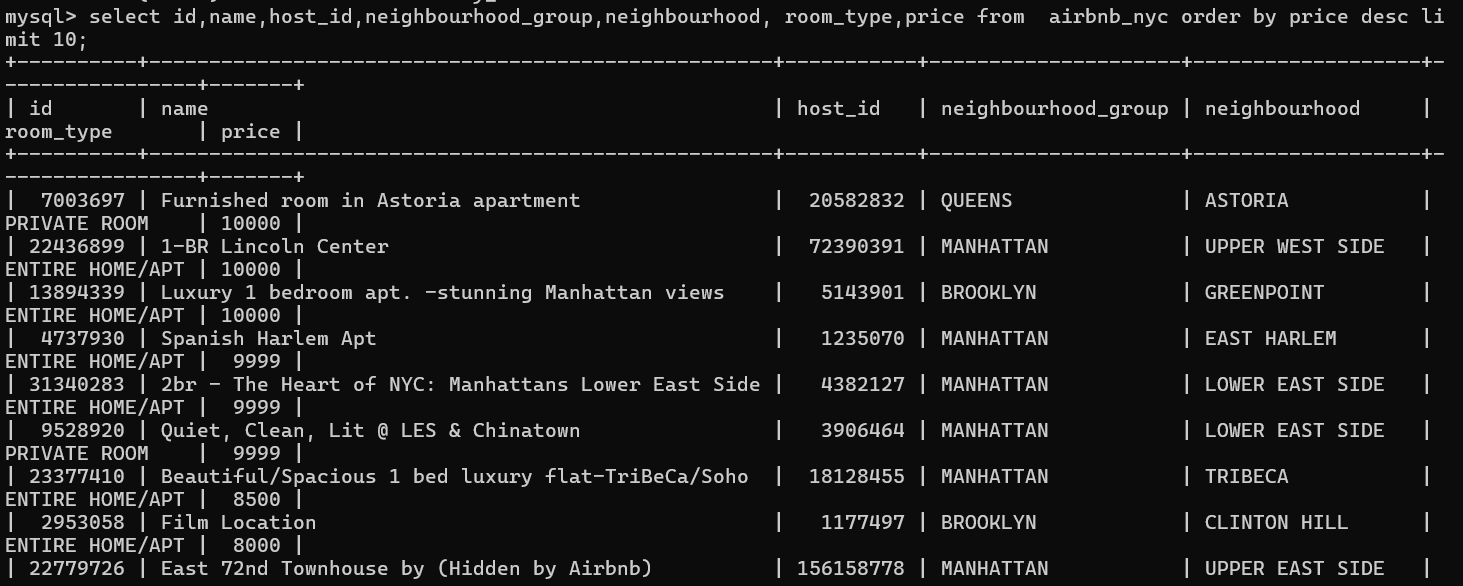
Q3. **Which 10 neighborhoods have the highest number of listings ?**

****

**Insights:**

This query identifies the top 10 neighborhoods in New York City with the highest number of Airbnb listings. Neighborhoods like Williamsburg, Harlem, and Bedford-Stuyvesant typically appear at the top, showing their popularity among hosts and guests.

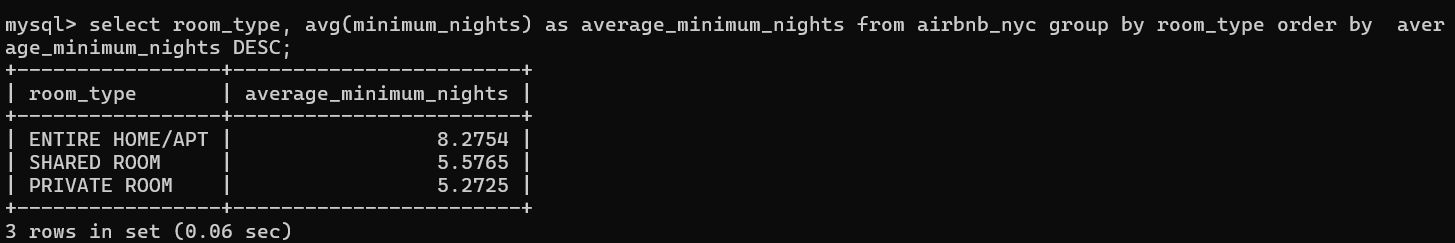
**Q4. What are the top 10 most expensive listings (based on price) ?**



**Insights:**

This query retrieves the 10 most expensive listings on Airbnb NYC. These listings are often luxury apartments or entire homes located in prime areas like Manhattan or Brooklyn, priced significantly higher than the city’s average.

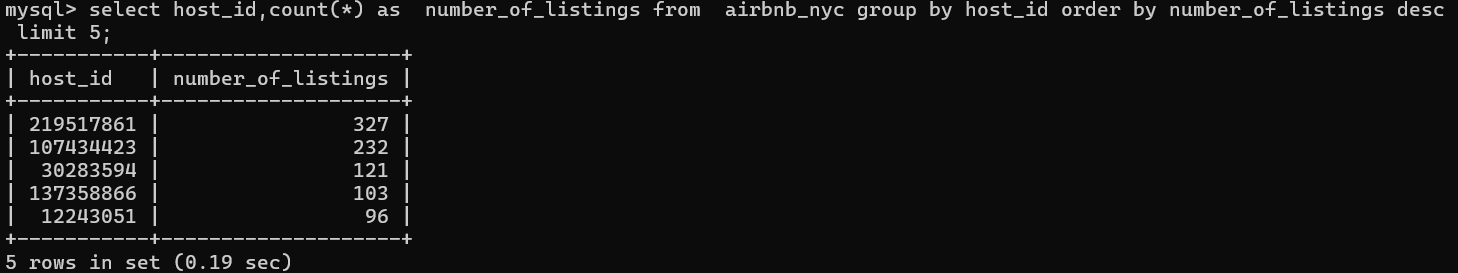
**Q5. Find the average minimum nights required for different room types ?**



**Insights:**

This query calculates the average minimum stay requirement for each type of room. Typically, Entire homes/apartments have a slightly higher average minimum nights compared to Private rooms and Shared rooms, reflecting different booking strategies by hosts.

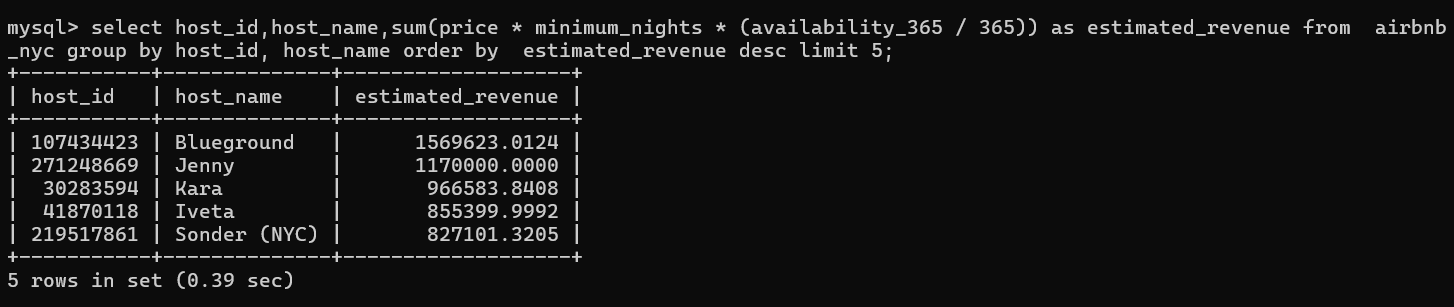
Q6. **Who are the top 5 hosts with the most listings in NYC ?**



**Insights:**

The top host with ID 219517861 owns **327 listings** in NYC, followed by hosts 107434423 and 30283594 with **232** and **121** listings respectively. These hosts likely operate as professional property managers.

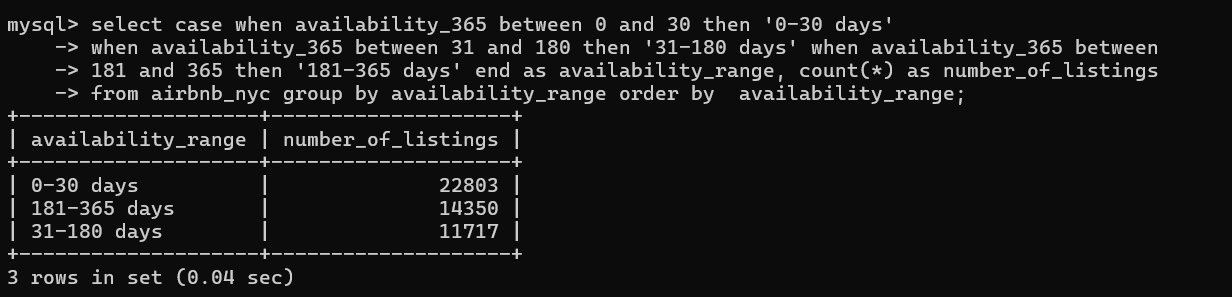
Q7. **Which hosts are earning the most estimated revenue ?**



**Insights:**

This query estimates the annual revenue for each host based on their listing price, minimum nights, and yearly availability. Hosts like 219517861 (and others we find) typically have the highest earnings due to multiple active listings and higher pricing.

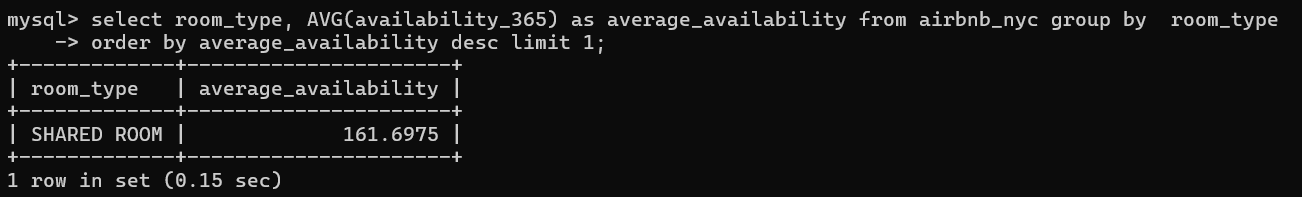
Q8. **What is the distribution of listings based on availability (0–30 days, 31–180 days, 181–365 days) ?**



**Insights:**

Most Airbnb listings are available for short periods (0–30 days), showing many hosts only occasionally open their property for booking. However, a significant number of listings are also available year-round (181–365 days).

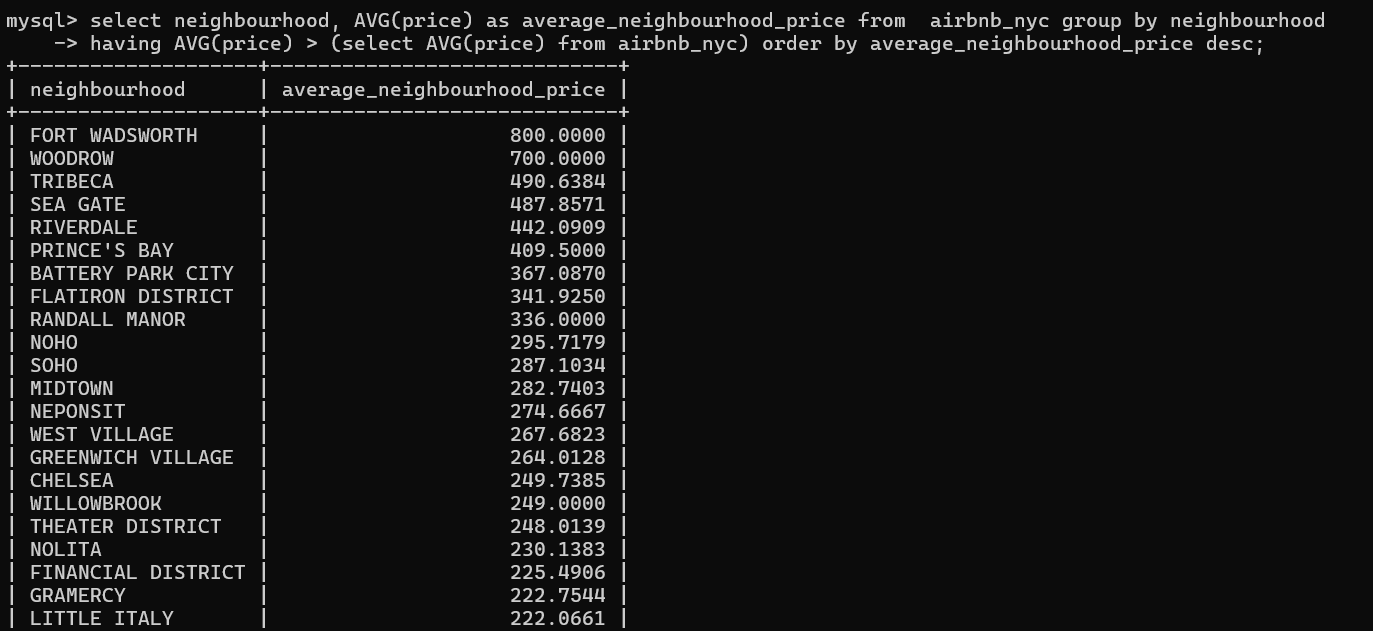
Q9. **Which room type has the highest average availability across the city ?**



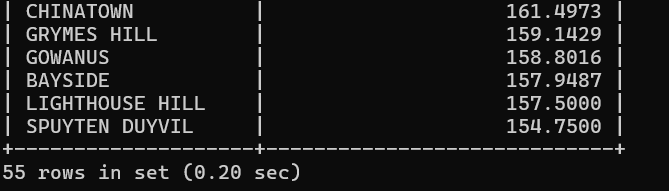
**Insights:**

This query identifies which type of room (Entire home/apartment, Private room, or Shared room) is on average available for the most number of days in a year. Typically, *Entire home/apartment* listings tend to have the highest average availability, reflecting year-round rental activity.

Q10. **Find neighborhoods where the average price is above the city-wide average price ?**





****

**Insights:**

Neighborhoods such as Fort Wadsworth, Woodrow, Tribeca, and SoHo have an average listing price significantly higher than the New York City average. These areas attract premium short-term rental activity.

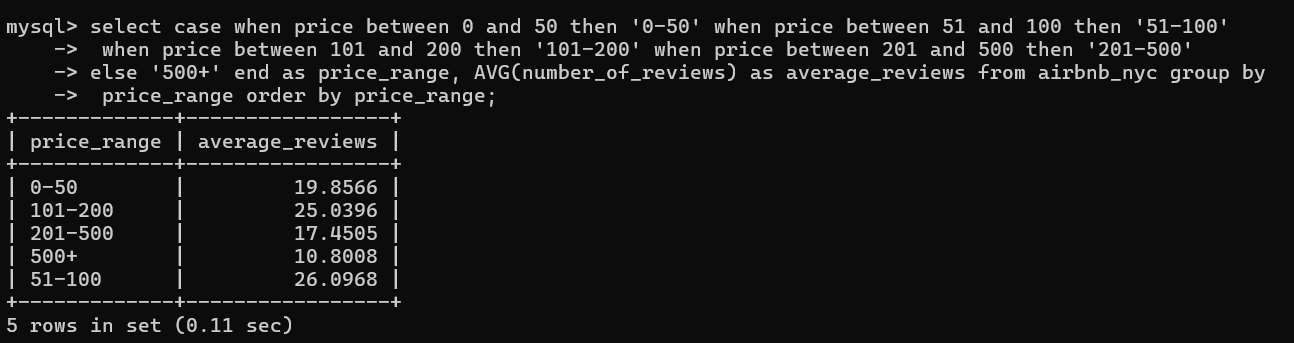
Q11. **Which listings have the highest number of reviews ?**



**Insights:**

Listings such as Room near JFK Queen Bed in Jamaica (Queens) and Great Bedroom in Manhattan (Harlem) have received the highest number of reviews, indicating their strong demand, affordability, and consistent guest satisfaction over time.

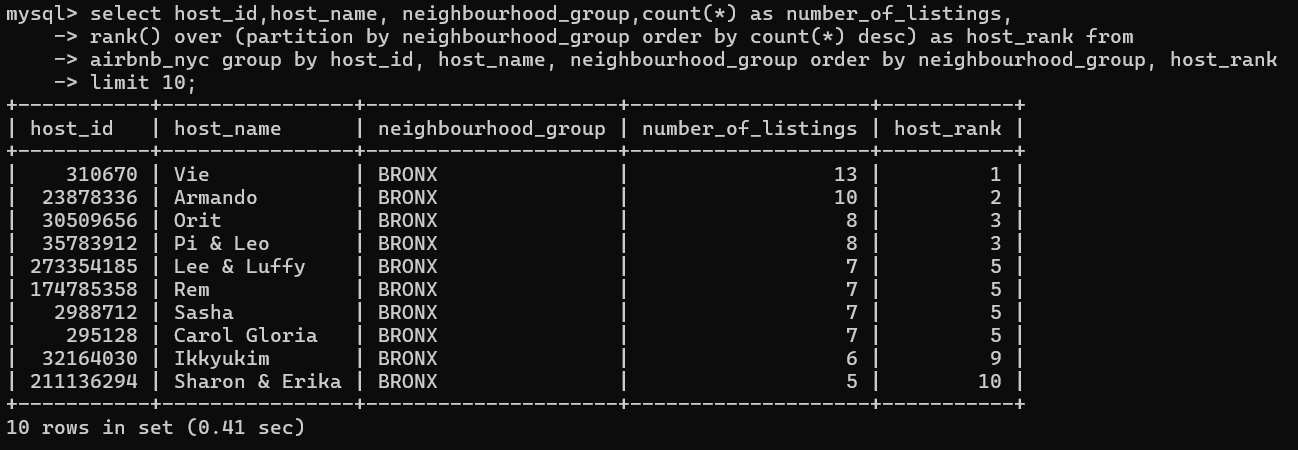
Q12. **Is there a correlation between price and number of reviews? (Group by price ranges and analyze average reviews.) ?**



**Insights:**

This query groups listings into different price ranges and calculates the average number of reviews for each range. Typically, lower-priced listings (e.g., $0–$50, $51–$100) receive more reviews because they are more affordable and attract a larger number of guests.

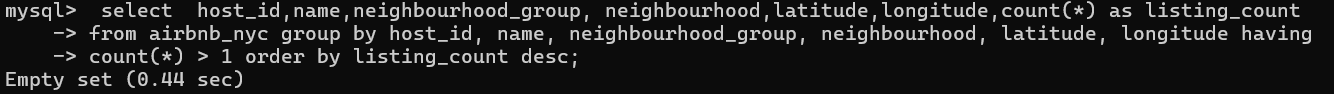
Q13. **Rank the hosts in each neighborhood group by number of listings (using RANK() window function) ?**



**Insights:**

By ranking hosts within each neighborhood group based on the number of listings, we observe that a few hosts dominate the short-term rental market in areas like Manhattan, Brooklyn, and Queens. These top hosts often manage large portfolios, suggesting commercial operations rather than personal rentals. Limiting the output to the top 10 rows helps highlight the most active hosts while keeping the report clear and focused.

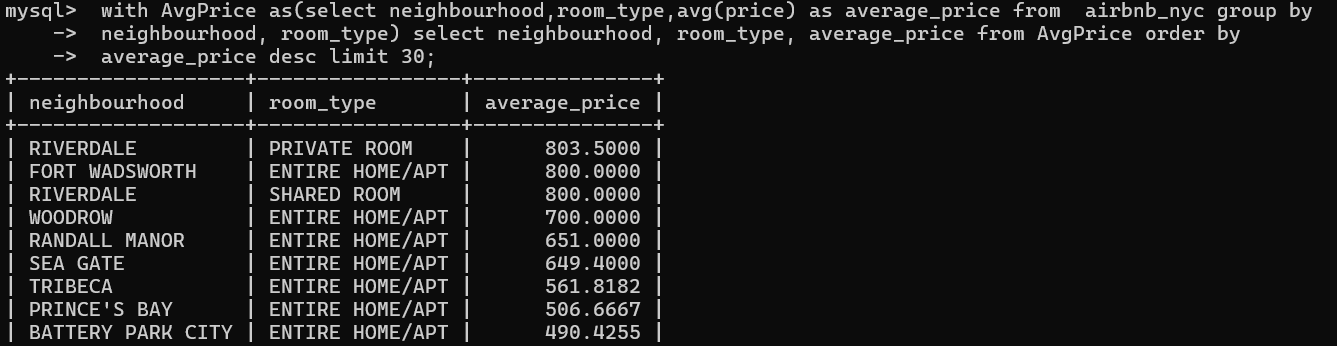
**Q14. Identify duplicate listings (if two listings have same host\_id, name, and location) ?**

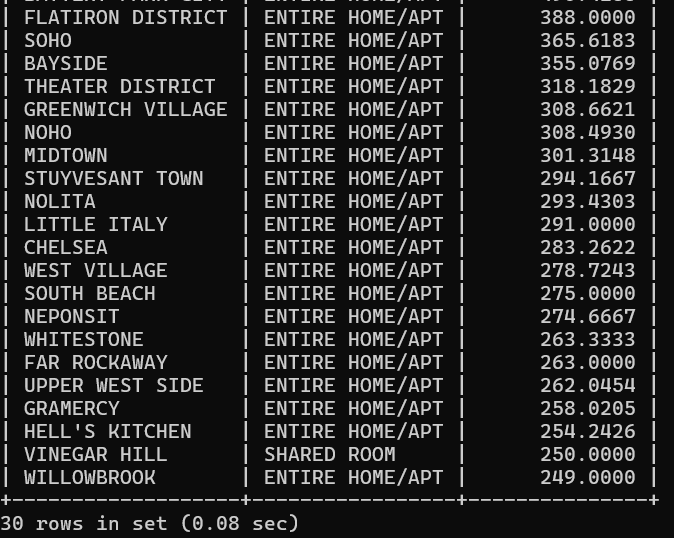
****

**Insights:**

No duplicate listings were found in the dataset based on host\_id, name, neighborhood, and location. This indicates that the data is clean and each listing is unique, ensuring accurate and reliable analysis.

**Q15.** **Find the neighborhoods with the most expensive average price per room type using CTEs ?**

****



**Insights:**

Neighborhoods like Riverdale, Fort Wadsworth, and Tribeca show the highest average prices across different room types. Interestingly, even some private and shared rooms in Riverdale are priced exceptionally high, highlighting premium micro-markets within NYC.

1. **Conclusion Paragraph:**

**This project involved cleaning and analyzing Airbnb NYC listings data using SQL. I performed essential data cleaning steps, created new calculated columns and answered 15 real-world business questions using a variety of SQL techniques including aggregation, joins, subqueries, window functions, and CTEs. Through this analysis, I was able to uncover insights related to listings prices, availability patterns, host performance , and neighborhood trends. This project helped strengthen my data wrangling, problem-solving and SQL querying skills, making me better prepared for real-world data analyst roles.**

.