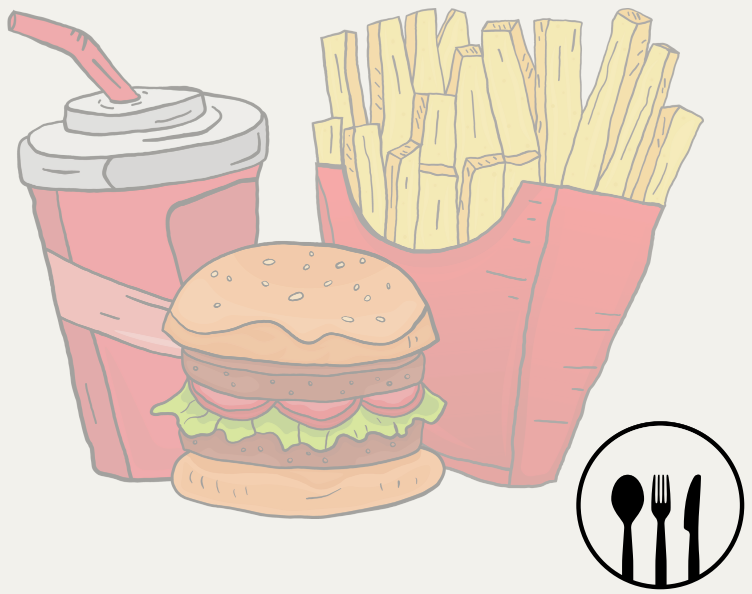
SQL Project

By

Suraj Maurya

Swiggy Restaurants Data Analysis



surajmaurya2

surajmaurya1

Project Aim

The aim of this project is to analyse Swiggy's restaurant data using **MySQL** to extract meaningful insights about customer preferences, restaurant performance, and delivery trends. This analysis will help identify patterns in cuisine popularity, pricing strategies, and delivery efficiencies, enabling data-driven decision-making to optimize restaurant operations, enhance customer satisfaction, and drive business growth.

Dataset Preparation and Enhancement

The dataset for this project was sourced from Kaggle, containing information about restaurants, cuisines, ratings, and related attributes. However, the original dataset did not include an orders table, which was essential for analyzing order trends and popular dishes. To address this, I created a custom table to simulate order-related data. Using Excel, I generated two new columns:

1. Total Orders: Representing the total number of orders placed at each restaurant, generated using the **RANDBETWEEN** formula to assign random values within a realistic range.
2. Popular Dishes: Listing the most frequently ordered dishes for each restaurant, derived from the available menu or cuisine data.

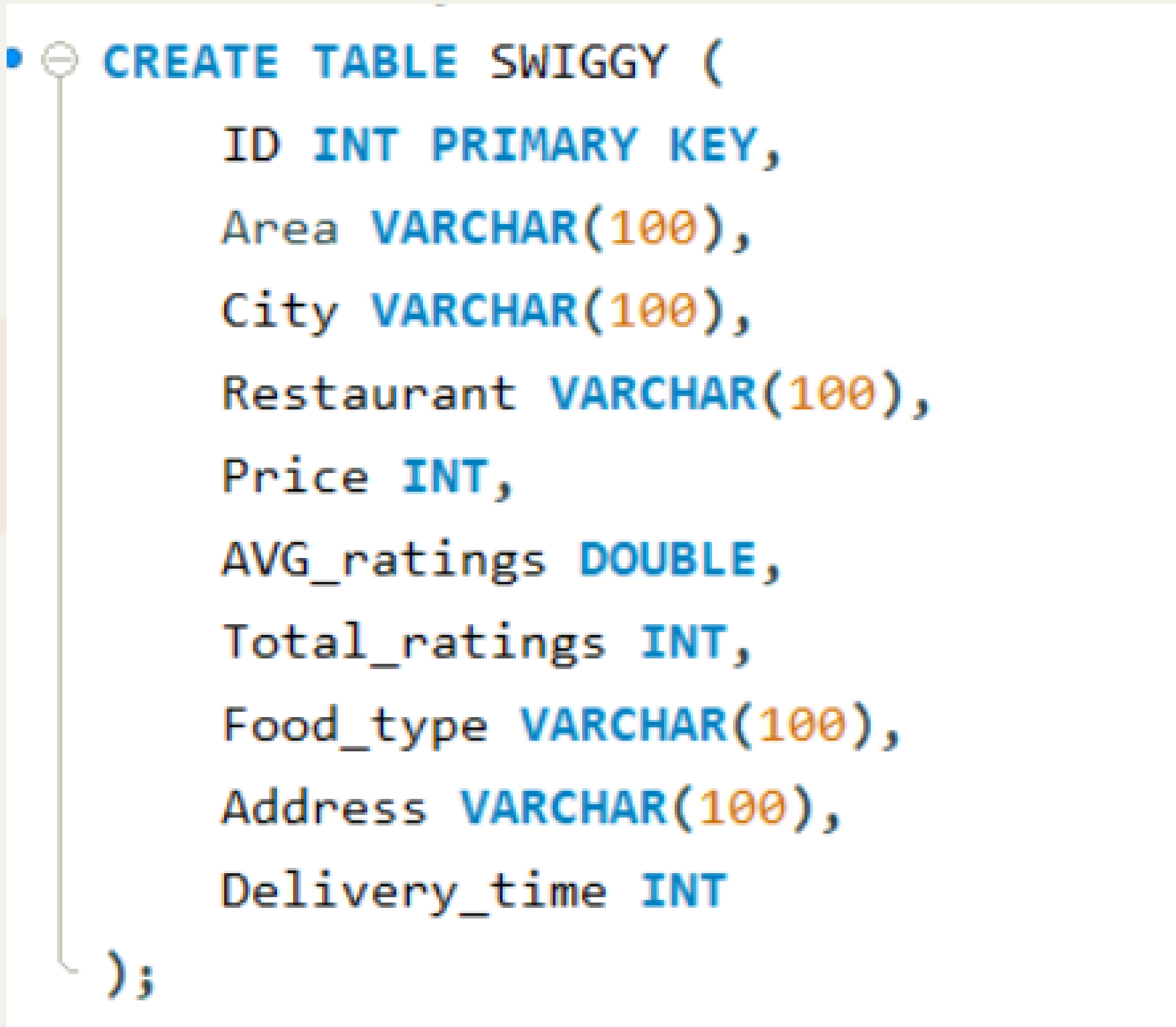
After generating these values, I converted the results into static values to ensure consistency and prevent recalculations. The enhanced table was then imported into the database as part of the dataset for analysis.

Kaggle dataset

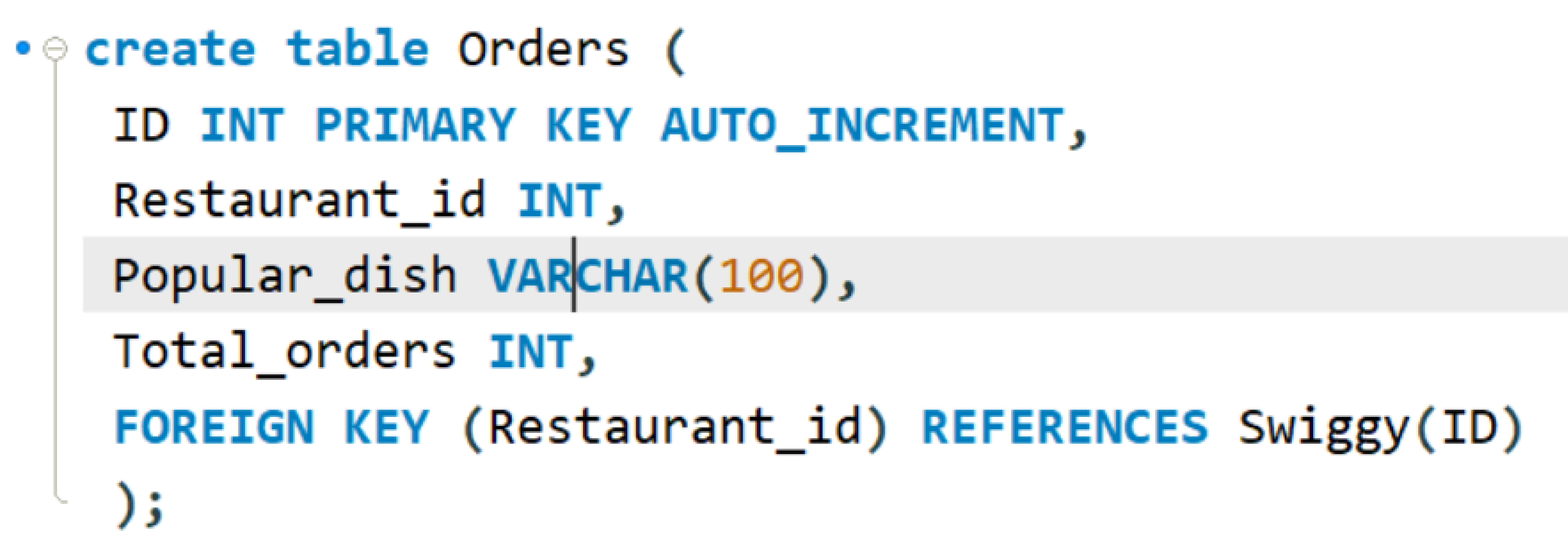
The Swiggy table stores restaurant information such as **ID**, **area**, **city**, **name**, **price**, **average ratings**, **total ratings**, **food type**, **address**, and **delivery time**.

The Orders table manages order details, including a **unique ID, associated restaurant ID** (foreign key referencing the **Swiggy table**), popular dishes, and total orders.

These tables work together to efficiently link orders to restaurants, track popular dishes, and manage delivery-related data in the food delivery system.



```
CREATE TABLE SWIGGY (
    ID INT PRIMARY KEY,
    Area VARCHAR(100),
    City VARCHAR(100),
    Restaurant VARCHAR(100),
    Price INT,
    AVG_ratings DOUBLE,
    Total_ratings INT,
    Food_type VARCHAR(100),
    Address VARCHAR(100),
    Delivery_time INT
);
```



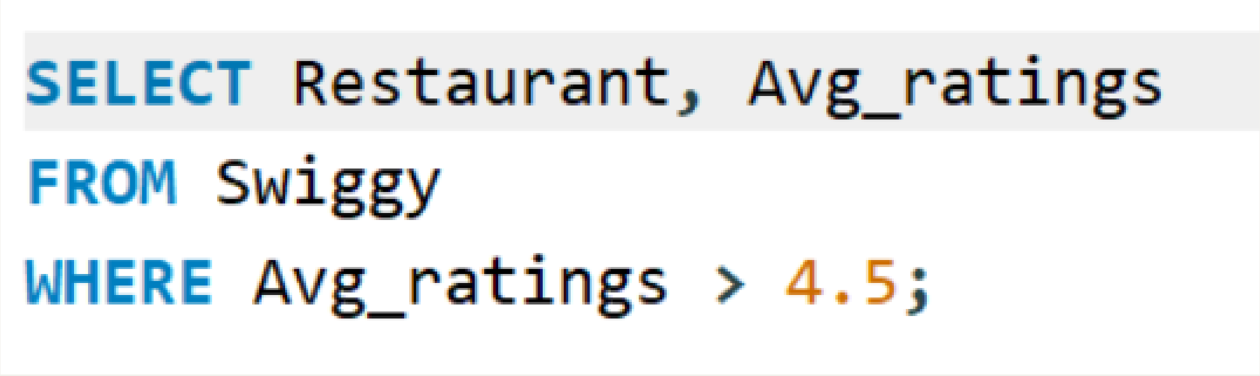
```
create table Orders (
 ID INT PRIMARY KEY AUTO_INCREMENT,
 Restaurant_id INT,
 Popular_dish VARCHAR(100),
 Total_orders INT,
 FOREIGN KEY (Restaurant_id) REFERENCES Swiggy(ID)
 );
```

Click to view Swiggy Table

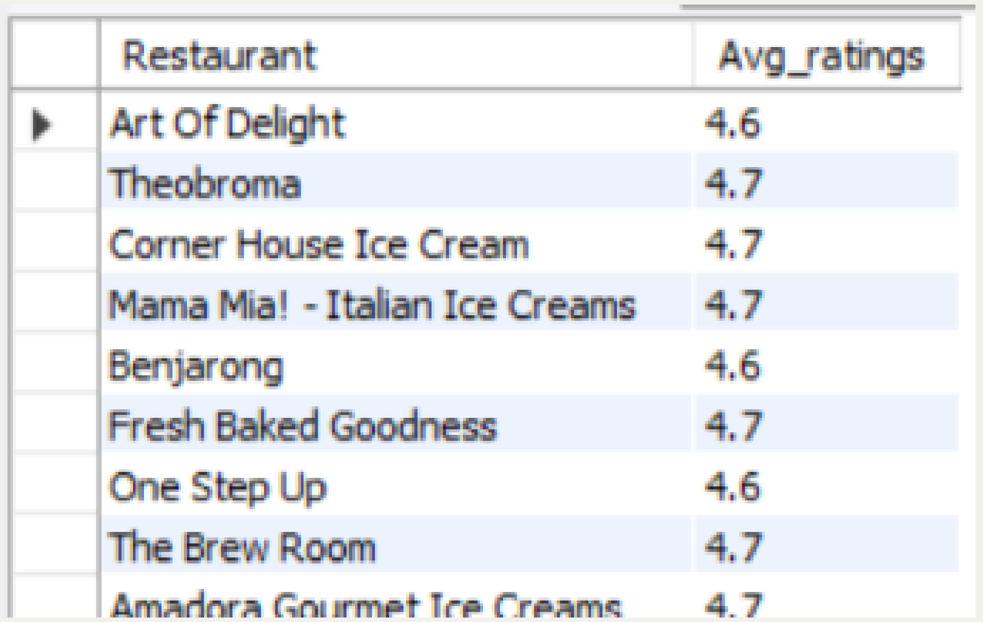
Click to view Orders Table

Analysis of High-Rated Restaurants

The objective of the SQL query in the image is to identify all restaurants with an average customer rating greater than **4.5**. This helps in pinpointing the best-performing restaurants based on customer feedback.



```sql
SELECT Restaurant, Avg_ratings
FROM Swiggy
WHERE Avg_ratings > 4.5;
```

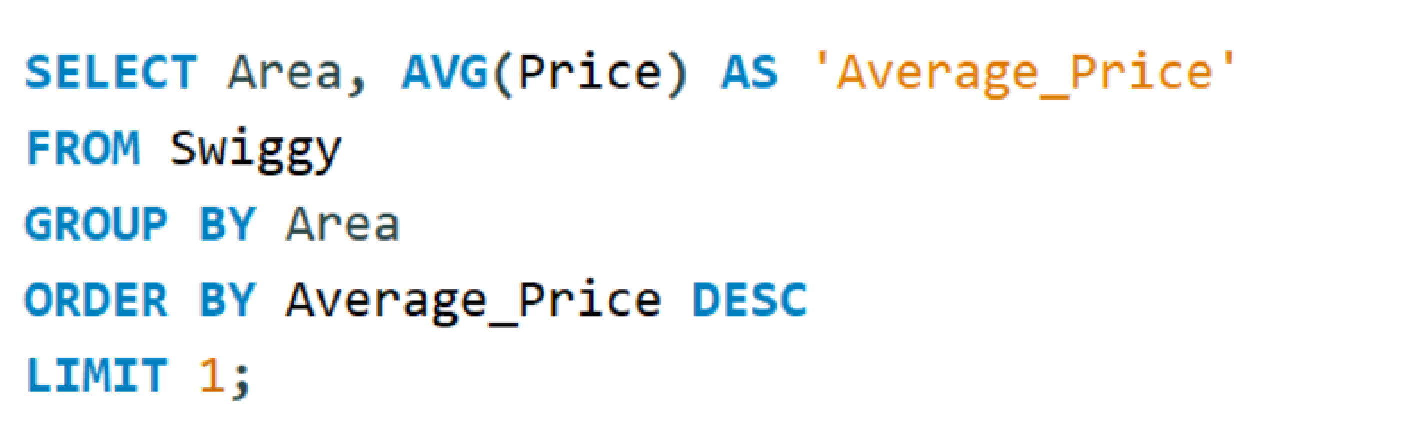


Restaurant	Avg_ratings
Art Of Delight	4.6
Theobroma	4.7
Corner House Ice Cream	4.7
Mama Mia! - Italian Ice Creams	4.7
Benjarong	4.6
Fresh Baked Goodness	4.7
One Step Up	4.6
The Brew Room	4.7
Amadora Gourmet Ice Creams	4.7

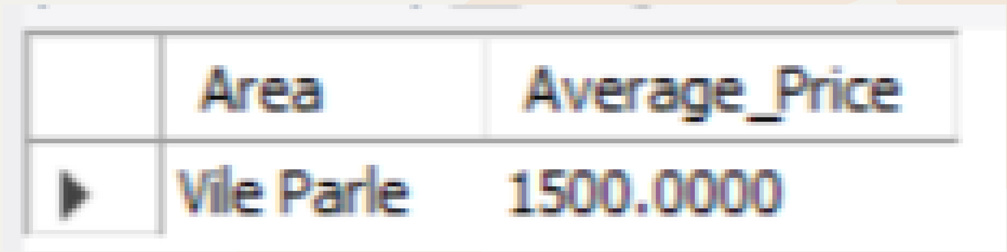
To view full result, Click here

Finding the Area with the Highest Average Price

The SQL query identifies the area with the highest average price for restaurant orders by calculating the average price of orders grouped by area. It sorts the results in descending order of average price and **limits** the output to a single result. The analysis revealed that **Vile Parle** has the highest average price of **₹1500**, suggesting it represents a premium market segment.



```
SELECT Area, AVG(Price) AS 'Average_Price'
FROM Swiggy
GROUP BY Area
ORDER BY Average_Price DESC
LIMIT 1;
```



	Area	Average_Price
▶	Vile Parle	1500.0000

Insights

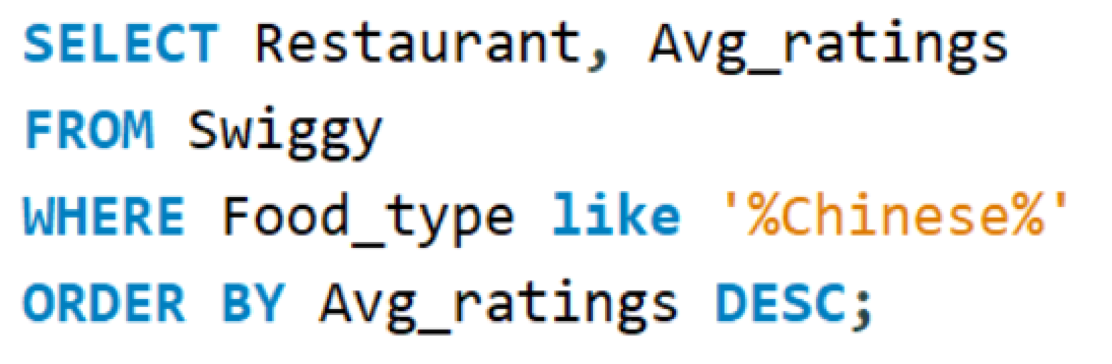
- Vile Parle appears to be a premium area where restaurant orders are more expensive than in other locations.
- This insight could guide business strategies, such as targeting premium restaurants in this area or offering exclusive deals.

Use Case:

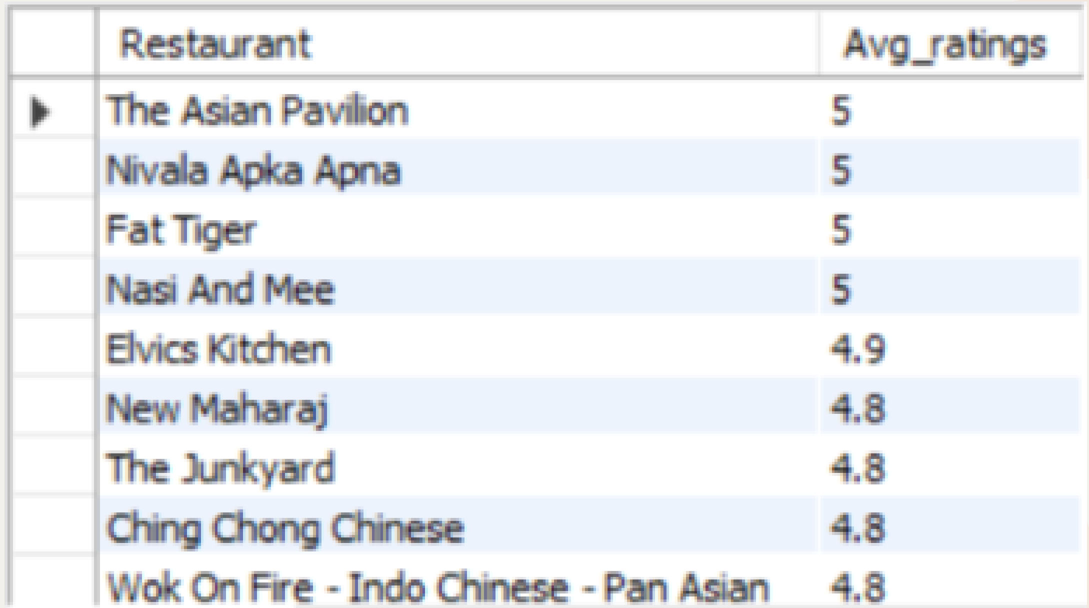
- Launch premium offerings or exclusive deals for high-end restaurants in this area.
- Expand partnerships with fine-dining establishments or upscale delivery options.
- Position Swiggy as the go-to platform for high-quality dining in affluent areas.

Analyzing Top-Rated Chinese Restaurants

The SQL query retrieves the names and average ratings of restaurants from the Swiggy table that serve Chinese cuisine. It filters the data using the **LIKE** operator to match any food type containing the word "**Chinese**" and sorts the results in descending order based on average ratings, ensuring that the highest-rated restaurants appear first. This query is designed to help identify the top-rated Chinese restaurants on the platform.



```sql
SELECT Restaurant, Avg_ratings
FROM Swiggy
WHERE Food_type like '%Chinese%'
ORDER BY Avg_ratings DESC;
```



	Restaurant	Avg_ratings
▶	The Asian Pavilion	5
	Nivala Apka Apna	5
	Fat Tiger	5
	Nasi And Mee	5
	Elvics Kitchen	4.9
	New Maharaj	4.8
	The Junkyard	4.8
	Ching Chong Chinese	4.8
	Wok On Fire - Indo Chinese - Pan Asian	4.8

To view full result, Click here

Insights:

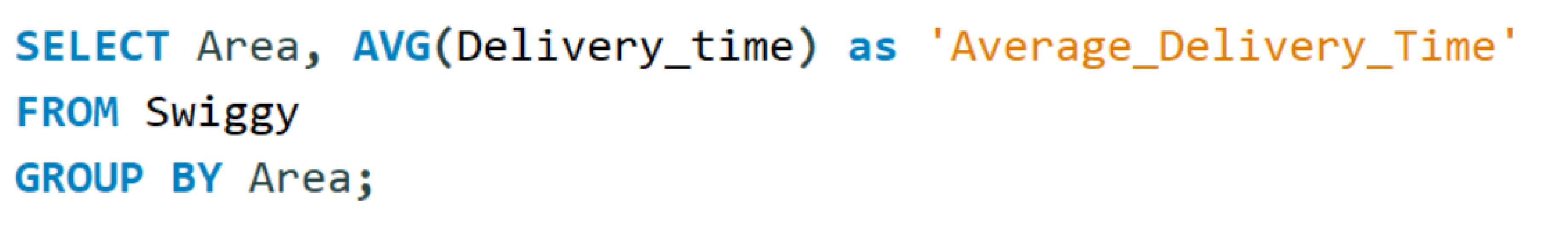
- Identifies top-rated Chinese restaurants on Swiggy with ratings from 2.0 to 5.0.
- Highlights strong competition and high customer satisfaction in this category.
- Restaurants like ***The Asian Pavilion*** and ***Nivala Apka Apna*** stand out with perfect ratings.

Use Case:

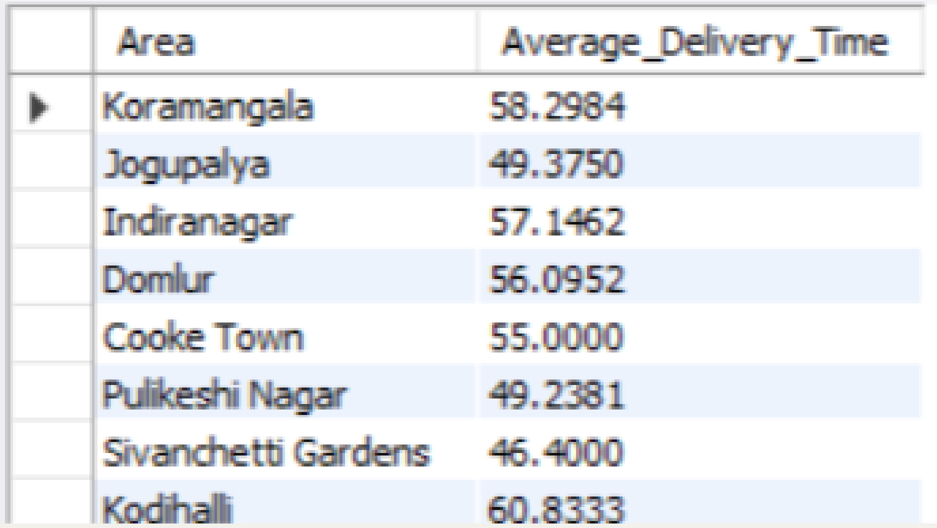
- Helps customers find the best-rated Chinese restaurants quickly.
- Restaurants with high ratings could be promoted in Swiggy campaigns, while those absent from the list can be encouraged to improve their offerings or service quality.

Analyzing Average Delivery Time by Area on Swiggy

The query selects the **Area** and computes the **average delivery time** (AVG(Delivery_time)) for each area in the dataset. It uses the **GROUP BY** clause to group the results by Area, ensuring that the average delivery time is calculated separately for each distinct area.



```
SELECT Area, AVG(Delivery_time) as 'Average_Delivery_Time'
FROM Swiggy
GROUP BY Area;
```



Area	Average_Delivery_Time
Koramangala	58.2984
Jogupalya	49.3750
Indiranagar	57.1462
Domlur	56.0952
Cooke Town	55.0000
Pulikeshi Nagar	49.2381
Sivanchetti Gardens	46.4000
Kodihalli	60.8333

To view full result, Click here

Insights:

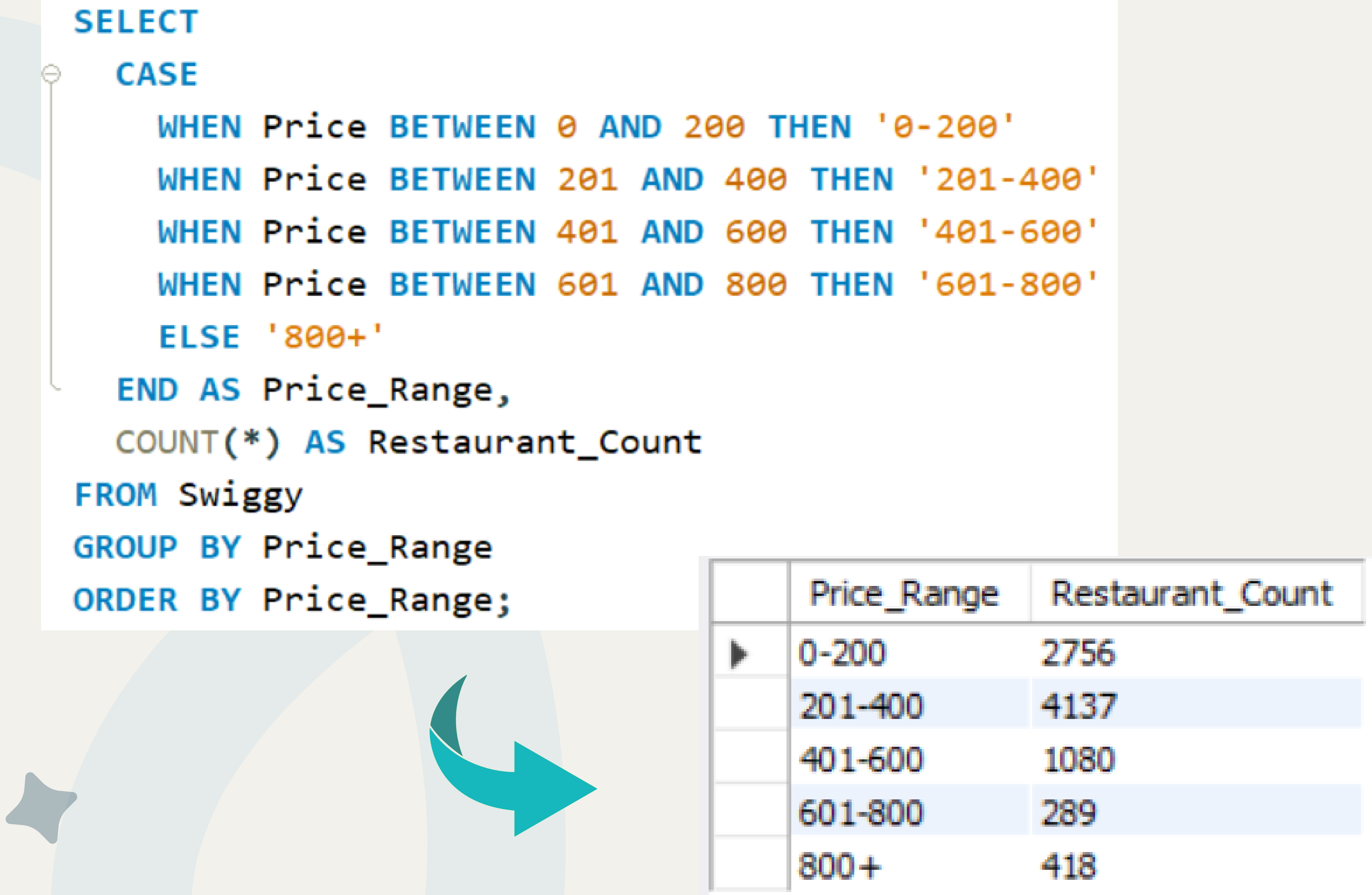
- The output shows the average delivery time for each area. For example, **Koramangala** has the highest average delivery time of 58.2984 minutes.
- We can use the **ORDER BY** clause to identify areas with the fastest and slowest average delivery times.
- Variations in delivery times could indicate differences in traffic, restaurant density, or operational efficiency across areas.

Use Case:

- **Operational Optimization:** Swiggy can use this data to improve delivery efficiency in high-delay areas.
- **Resource Allocation:** Swiggy can allocate more delivery agents or optimize routes in areas with higher average delivery times, such as Koramangala.

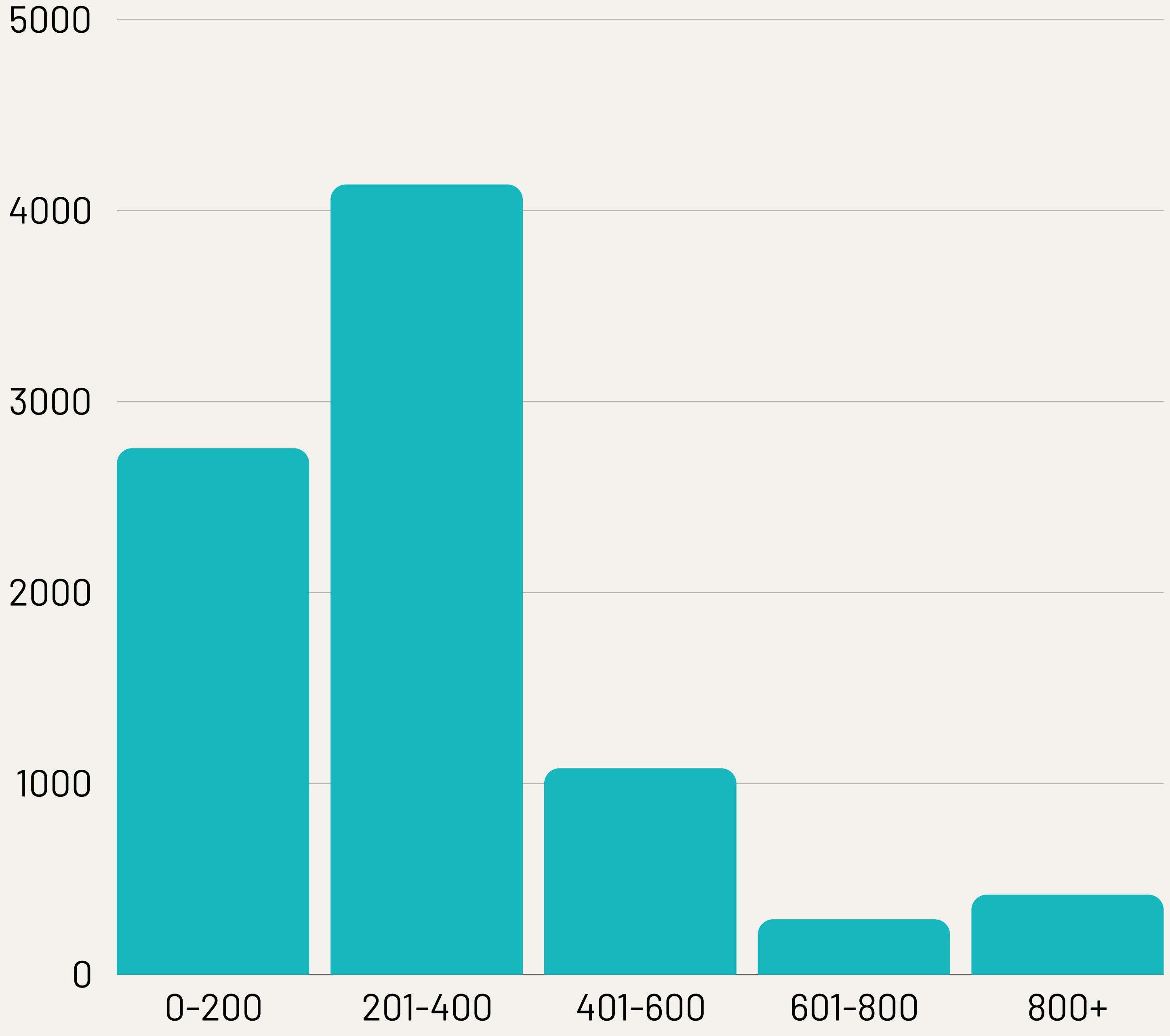
Restaurant Count Analysis Across Price Ranges

The SQL query categorizes restaurants from the swiggy table using **CASE** expression into price ranges (**0-200, 201-400, 401-600, 601-800, 800+**) and counts the number of restaurants in each range. The output reveals that the most common price range is 201-400 with **4,137** restaurants, followed by 0-200 with **2,756**. High-end options (800+) account for **418** restaurants, while the 601-800 range has the fewest at **289**.



```
SELECT
  CASE
    WHEN Price BETWEEN 0 AND 200 THEN '0-200'
    WHEN Price BETWEEN 201 AND 400 THEN '201-400'
    WHEN Price BETWEEN 401 AND 600 THEN '401-600'
    WHEN Price BETWEEN 601 AND 800 THEN '601-800'
    ELSE '800+'
  END AS Price_Range,
  COUNT(*) AS Restaurant_Count
FROM Swiggy
GROUP BY Price_Range
ORDER BY Price_Range;
```

Price_Range	Restaurant_Count
0-200	2756
201-400	4137
401-600	1080
601-800	289
800+	418



Highlights

Key Insights:

- The 201-400 price range dominates with 4,137 restaurants, followed by the budget-friendly 0-200 range with 2,756.
- High-end restaurants (800+) account for just 418, while mid-tier (601-800) has the fewest with 289.
- Over 60% of restaurants fall in the price ranges of 0-400, indicating a significant focus on affordability.

Highlight:

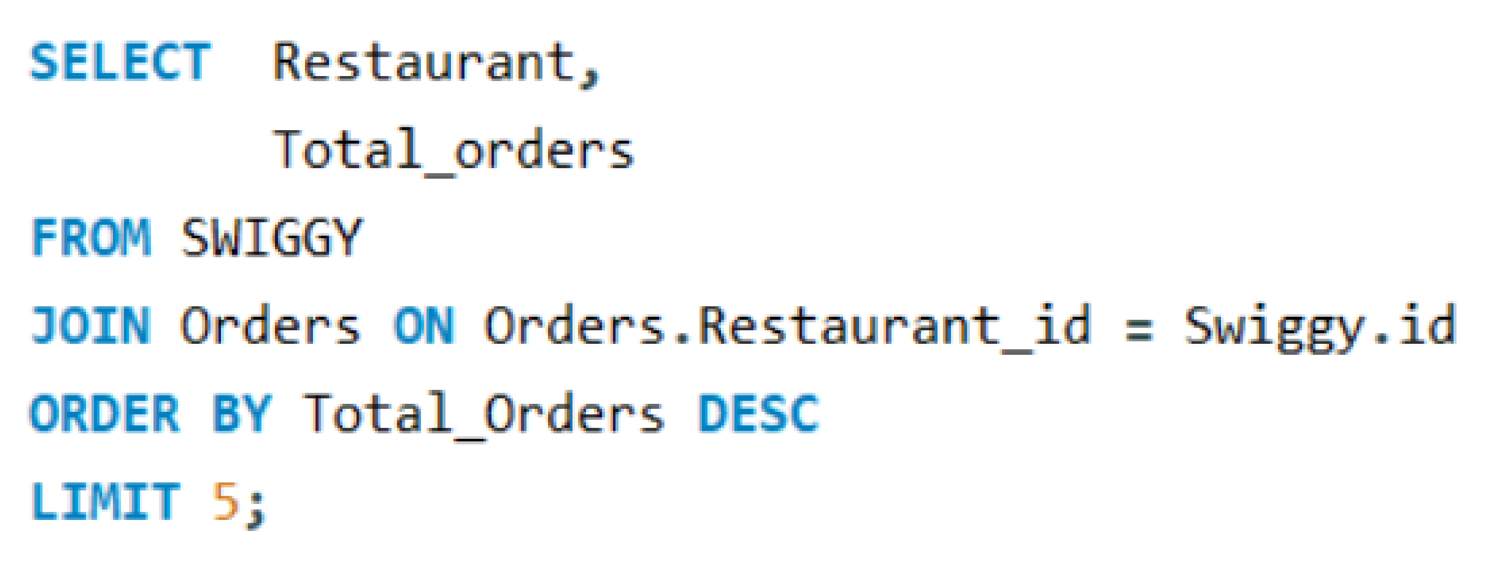
- Restaurants in the 201-400 range outnumber those in the 601-800 range by over 14 times, highlighting a gap in mid-tier options.

Recommendations:

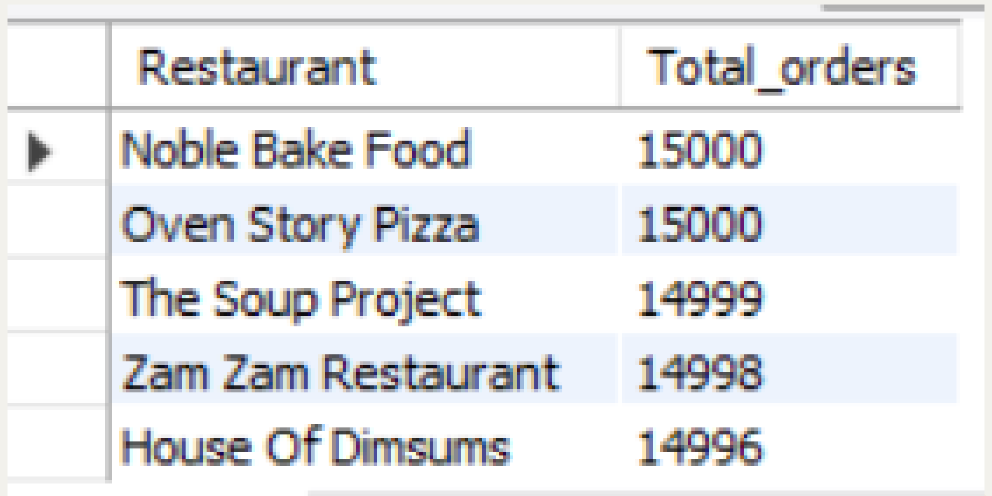
- Customer Focus: Design promotions or discounts targeting the highly competitive 201-400 range to attract more customers.
- Market Gaps: Focus on onboarding more restaurants in the 601-800 range to fill the mid-tier gap.
- High-End Appeal: Enhance visibility for premium (800+) options to capture affluent customers.

Top 5 Most-Ordered Restaurants on Swiggy

The query retrieves the top 5 restaurants with the highest total orders from the SWIGGY and Orders tables by performing a **join** on **Restaurant_id**. Results are sorted in descending order by **Total_orders**, with Noble Bake Food and Oven Story Pizza leading at 15,000 orders each.



```
SELECT  Restaurant,
        Total_orders
FROM SWIGGY
JOIN Orders ON Orders.Restaurant_id = Swiggy.id
ORDER BY Total_Orders DESC
LIMIT 5;
```



	Restaurant	Total_orders
▶	Noble Bake Food	15000
	Oven Story Pizza	15000
	The Soup Project	14999
	Zam Zam Restaurant	14998
	House Of Dimsums	14996

Data is for illustrative purposes only and may not reflect actual trends, as Total_orders were manually assigned for demonstration.

Insights:

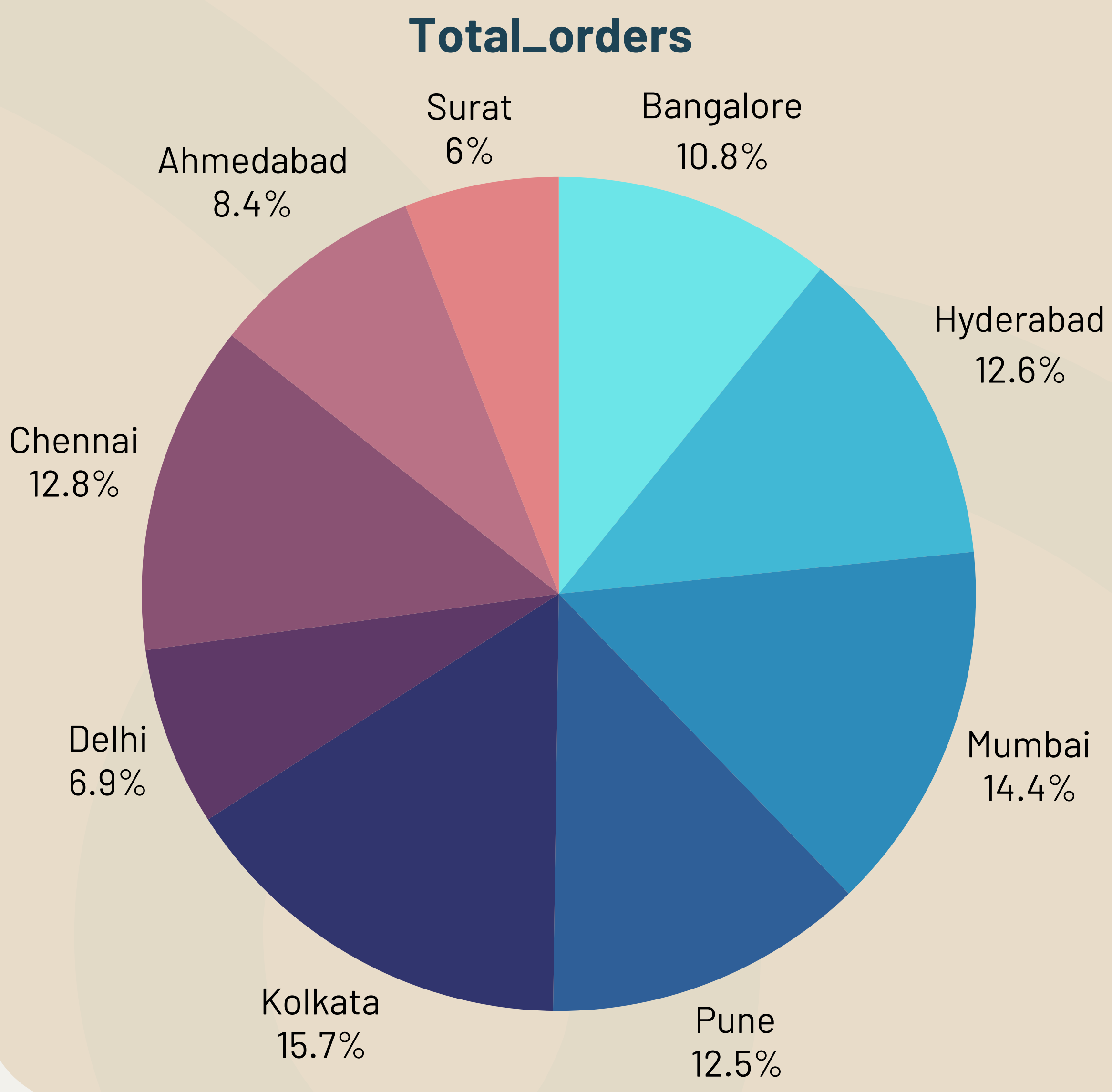
- Noble Bake Food and Oven Story Pizza lead the list with 15,000 orders each, showcasing their high popularity among users.
- These top-performing restaurants likely have a combination of efficient service, popular menu offerings, and strong customer loyalty driving high order volumes.

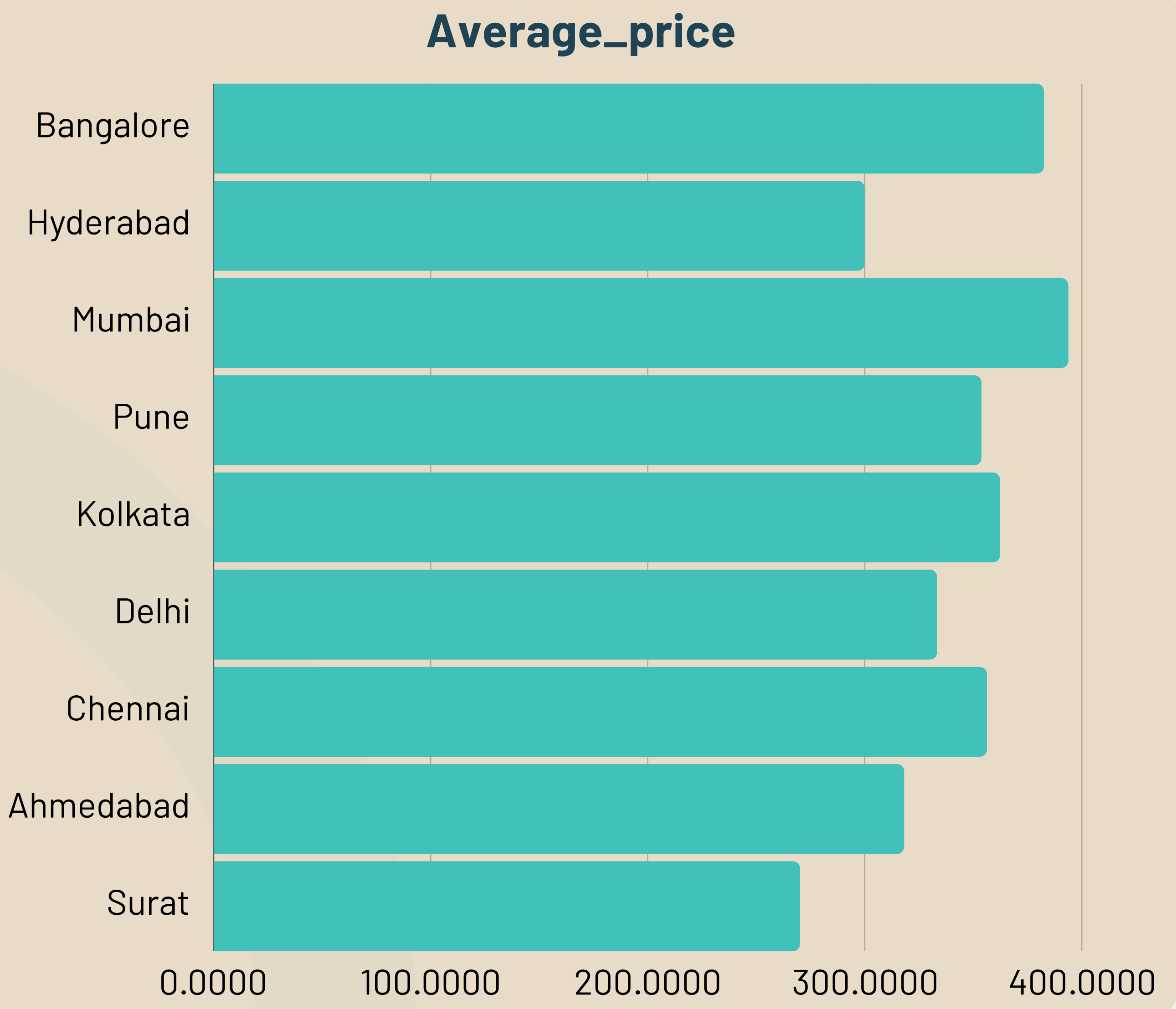
Use Case:

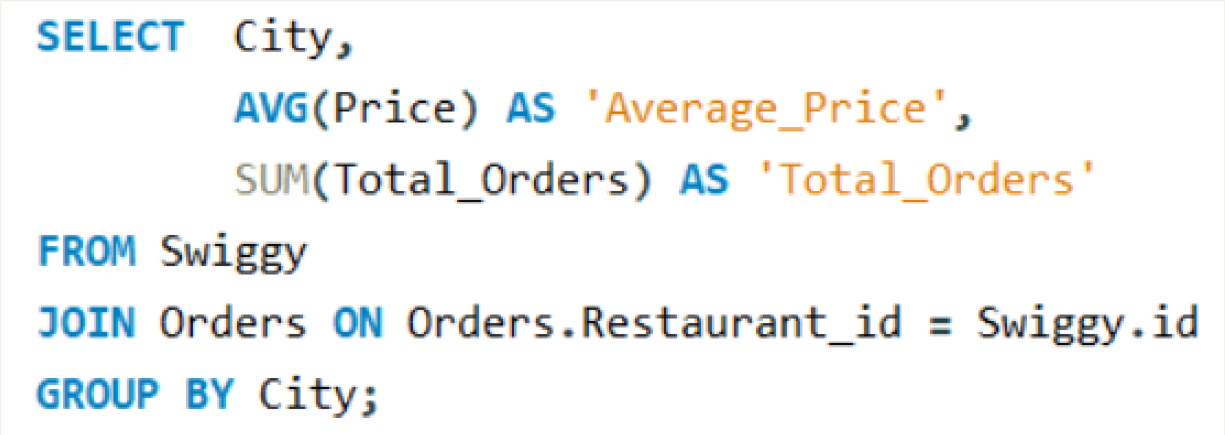
- **Operational Excellence:** Swiggy can analyze the practices of these top-performing restaurants (menu selection, pricing, delivery efficiency) to replicate success across other restaurants.
- **Data-Driven Partnerships:** Highlight these restaurants in campaigns or app banners to capitalize on their popularity and enhance user engagement.

City-Wise Analysis of Pricing and Order Volumes on Swiggy

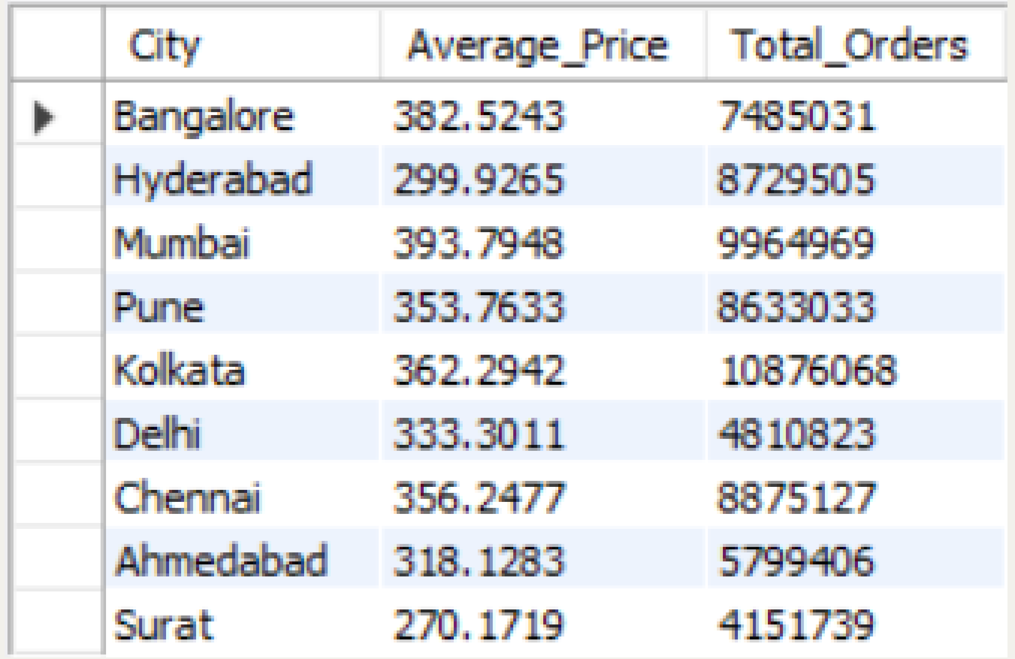
This updated query calculates the average price (AVG(Price)) and total orders (SUM(Total_Orders) grouped by city by joining the Swiggy and Orders tables. The results show Mumbai has the highest average price (393.79), while Kolkata leads with the most total orders (10,876,068). The data spans multiple cities, revealing key trends in pricing and order volume.







```
SELECT  City,
        AVG(Price) AS 'Average_Price',
        SUM(Total_Orders) AS 'Total_Orders'
FROM Swiggy
JOIN Orders ON Orders.Restaurant_id = Swiggy.id
GROUP BY City;
```



City	Average_Price	Total_Orders
Bangalore	382.5243	7485031
Hyderabad	299.9265	8729505
Mumbai	393.7948	9964969
Pune	353.7633	8633033
Kolkata	362.2942	10876068
Delhi	333.3011	4810823
Chennai	356.2477	8875127
Ahmedabad	318.1283	5799406
Surat	270.1719	4151739

Highlights

Insights:

Kolkata: Highest Total Orders.
Mumbai: Highest Average Price.
Surat: Lowest Average Price (₹270.17).
Pune and Chennai balance affordability (₹350–₹360) and high order volumes.

Use Cases

Business Growth:
Boost marketing in top-performing cities (e.g., Kolkata, Hyderabad).
Launch discount campaigns in cities with lower order volumes (e.g., Surat, Ahmedabad).

Resource Optimization:
Allocate delivery personnel and inventory based on total order volumes.

Pricing Strategy:
Leverage insights on average prices to design city-specific offers.

Restaurant Partnerships:
Partner with premium restaurants in cities with high average prices like Mumbai.
Encourage affordable dining in cities like Surat to increase orders.

Recommendations

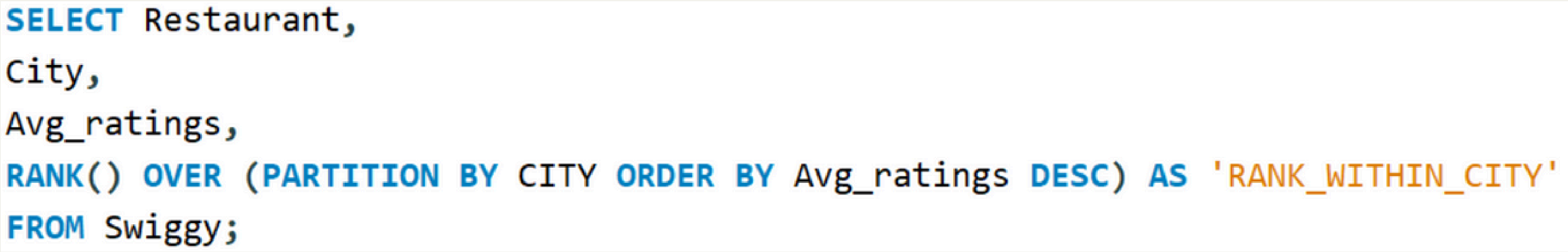
Marketing Campaigns:
High-order cities: Retain customers with loyalty programs.
Low-order cities: Attract customers with targeted discounts.

Price Optimization:
Analyse customer behaviour for cities with low order volumes and tweak pricing strategies accordingly.

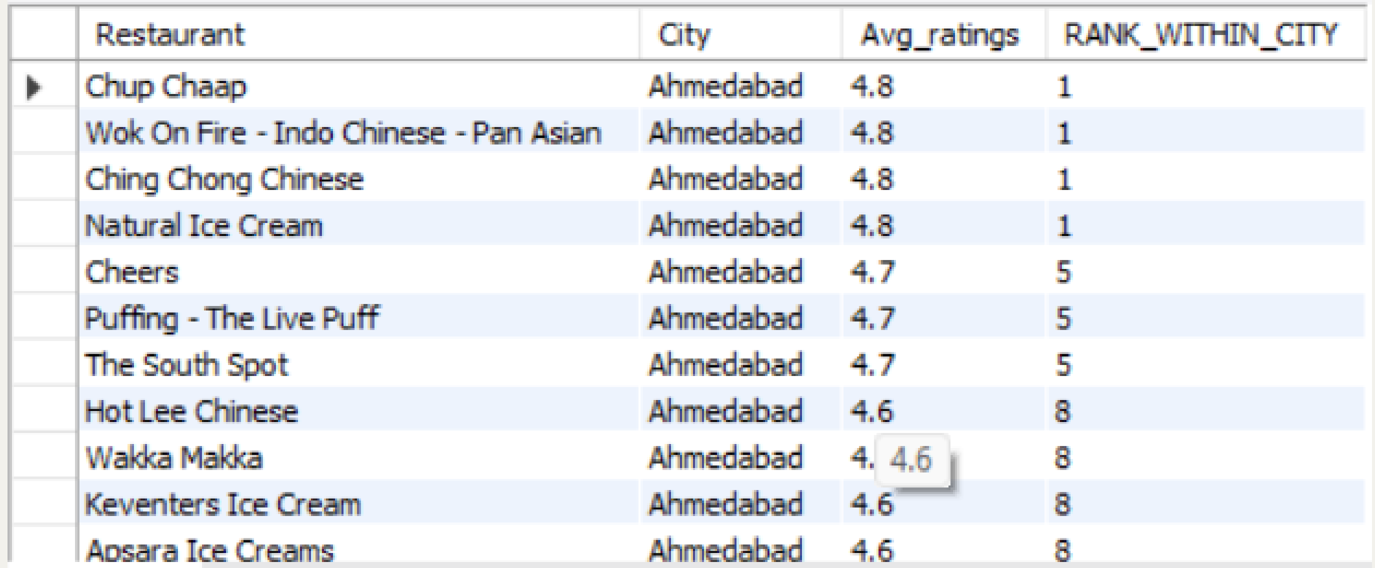
Expand Partnerships:
Collaborate with local and premium restaurants for city-specific needs.

Restaurant Rankings and Ratings Across Cities

The SQL query partitions restaurant data by city and assigns a rank to each restaurant based on its average rating within that city. The result allows identification of the top-rated restaurants per city, enabling businesses or users to prioritize dining options. Ties in ratings share the same rank, ensuring an unbiased ranking system.



```sql
SELECT Restaurant,
City,
Avg_ratings,
RANK() OVER (PARTITION BY CITY ORDER BY Avg_ratings DESC) AS 'RANK_WITHIN_CITY'
FROM Swiggy;
```



Restaurant	City	Avg_ratings	RANK_WITHIN_CITY
Chup Chaap	Ahmedabad	4.8	1
Wok On Fire - Indo Chinese - Pan Asian	Ahmedabad	4.8	1
Ching Chong Chinese	Ahmedabad	4.8	1
Natural Ice Cream	Ahmedabad	4.8	1
Cheers	Ahmedabad	4.7	5
Puffing - The Live Puff	Ahmedabad	4.7	5
The South Spot	Ahmedabad	4.7	5
Hot Lee Chinese	Ahmedabad	4.6	8
Wakka Makka	Ahmedabad	4.6	8
Keventers Ice Cream	Ahmedabad	4.6	8
Apsara Ice Creams	Ahmedabad	4.6	8

To view full result, Click here

Insights:

Top Performers: Restaurants like "Chup Chaap," "Wok On Fire," and "Ching Chong Chinese" in Ahmedabad share the highest average rating of 4.8 and rank first within the city.

Drop in Ratings: There is a noticeable drop from 4.8 to 4.7 (rank 5) and then to 4.6 (rank 8), showing a significant cutoff in customer perception among the top-rated and other restaurants.

City-Specific Insights: This ranking is city-specific, ensuring that the ratings reflect local customer preferences, which can guide businesses in competitive improvements.

Use Case

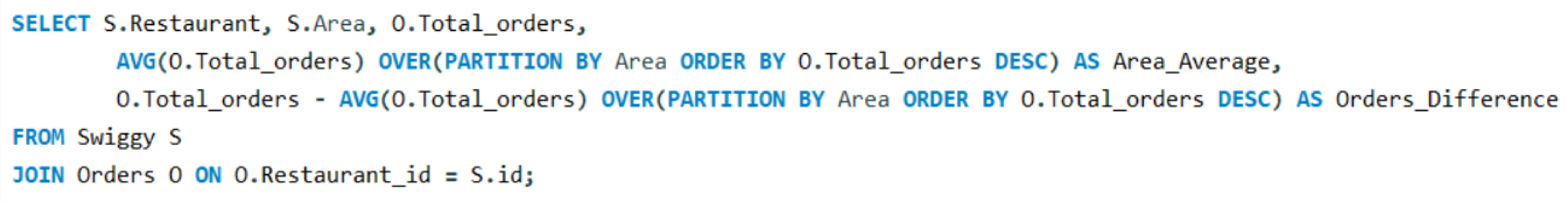
Customer Recommendations: Suggesting the top-rated restaurants to users based on their location.

Promotions: Highlighting the best-performing restaurants for marketing campaigns or featured listings.

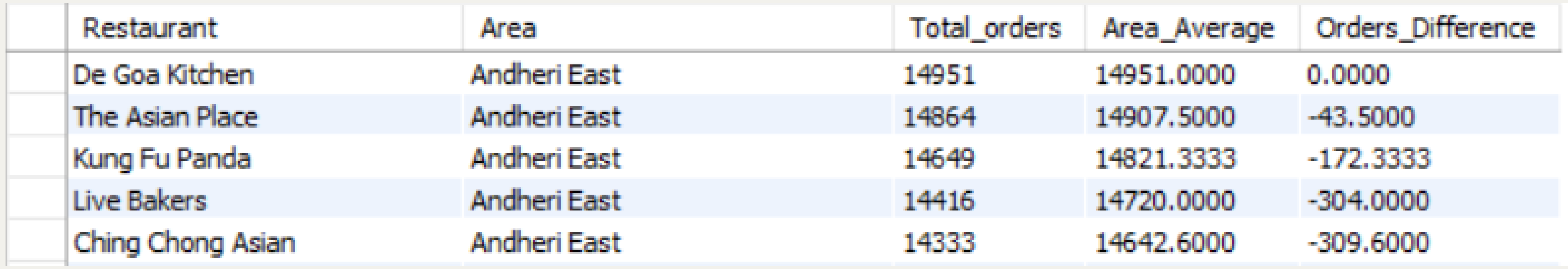
Strategic Partnerships: Prioritizing collaboration with highly rated restaurants.

Restaurant Performance Against Area Averages in Total Orders

The SQL query analyzes restaurant performance by calculating the **average total orders** (Area_Average) within each area and the deviation of each restaurant's total orders from this average (Orders_Difference). Using window functions, it partitions data by area and ranks restaurants based on their total orders. The output showcases how each restaurant's total orders compare to the average for their area, providing insights into which restaurants perform above or below the local average, helping identify top and underperforming outlets.



```
SELECT S.Restaurant, S.Area, O.Total_orders,
       AVG(O.Total_orders) OVER(PARTITION BY Area ORDER BY O.Total_orders DESC) AS Area_Average,
       O.Total_orders - AVG(O.Total_orders) OVER(PARTITION BY Area ORDER BY O.Total_orders DESC) AS Orders_Difference
FROM Swiggy S
JOIN Orders O ON O.Restaurant_id = S.id;
```



Restaurant	Area	Total_orders	Area_Average	Orders_Difference
De Goa Kitchen	Andheri East	14951	14951.0000	0.0000
The Asian Place	Andheri East	14864	14907.5000	-43.5000
Kung Fu Panda	Andheri East	14649	14821.3333	-172.3333
Live Bakers	Andheri East	14416	14720.0000	-304.0000
Ching Chong Asian	Andheri East	14333	14642.6000	-309.6000

Use Case: Analysing total orders of restaurants in a specific area to compute area averages and the difference in orders compared to the area's average.

Recommendation:

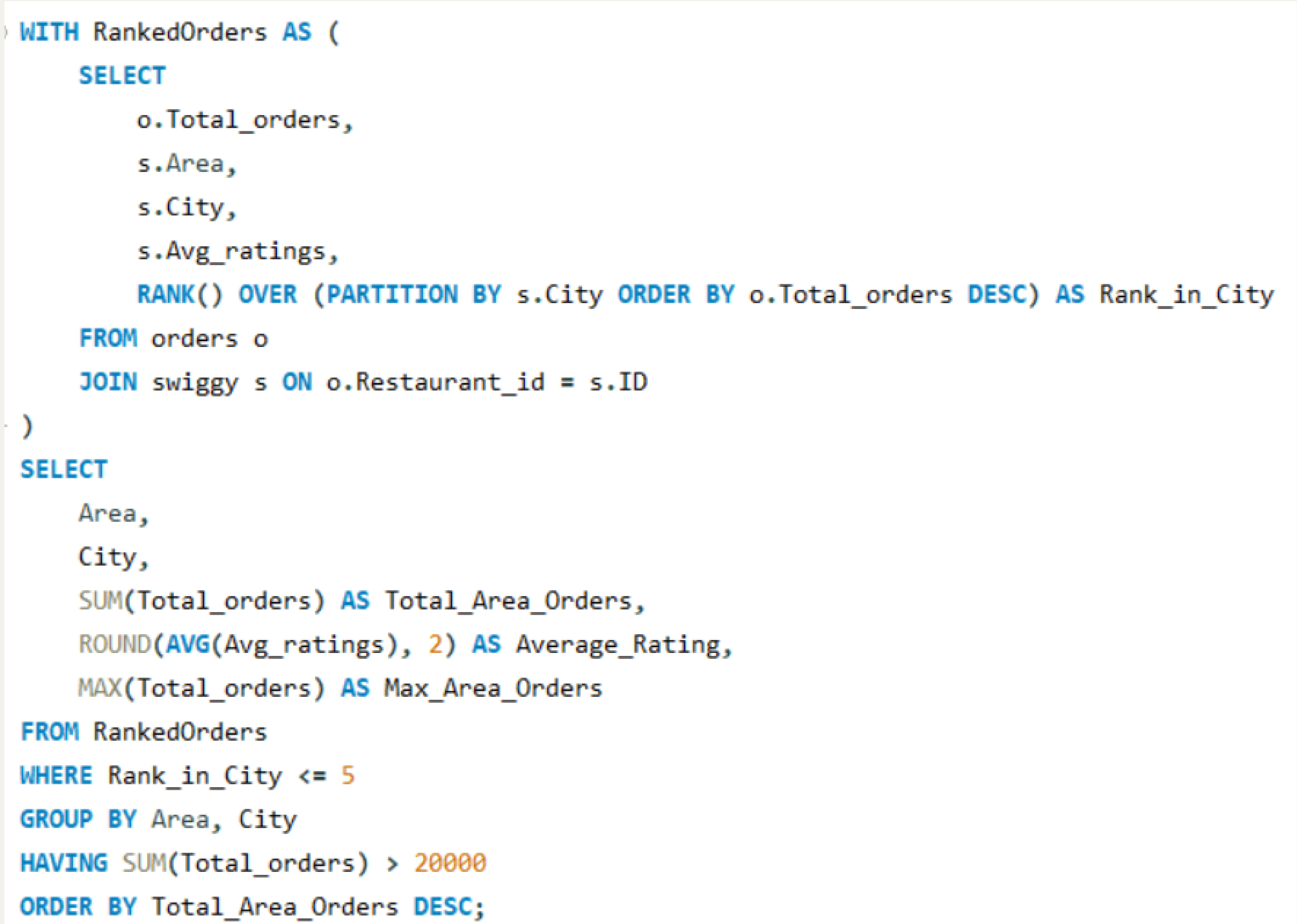
For Underperforming Restaurants: Focus on targeted marketing campaigns, menu optimization, or customer engagement to close the gap.

For Top Performers: Reinforce strengths through loyalty programs or enhanced visibility on platforms.

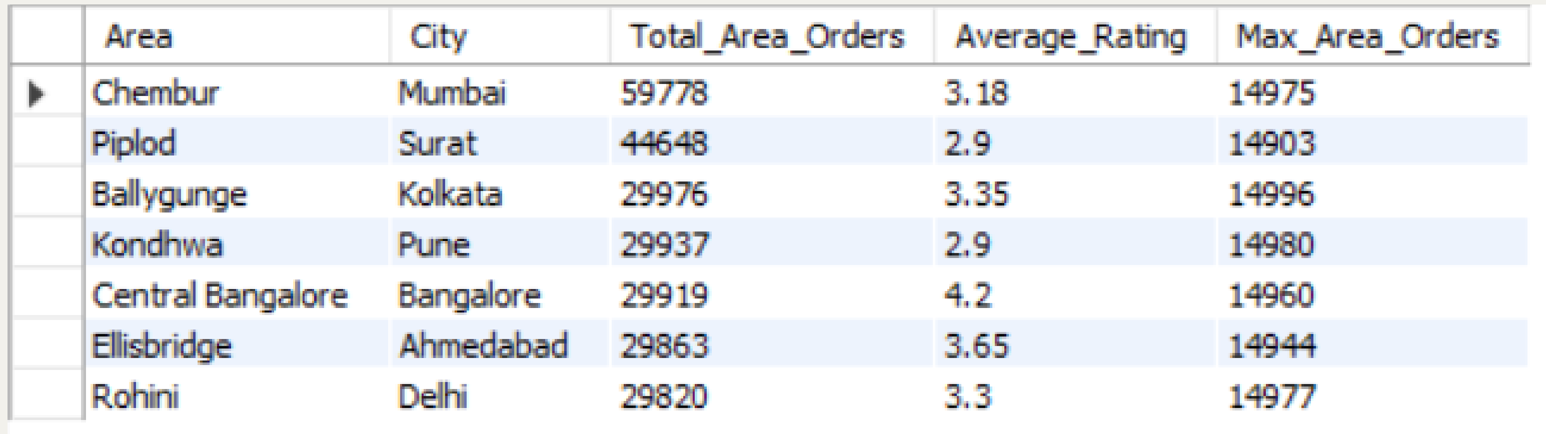
To view full result, Click here

Top Restaurants and High-Demand Areas by City

This query utilizes a **WITH** clause to define a temporary dataset where restaurants are ranked using the **RANK** window function. It filters the top 5 per city, calculates **SUM**, **MAX**, and **AVG** of orders and ratings, groups data by area and city, and applies a **HAVING** condition for areas with over **20000** orders, and sorts results by total demand.



```sql
WITH RankedOrders AS (
    SELECT
        o.Total_orders,
        s.Area,
        s.City,
        s.Avg_ratings,
        RANK() OVER (PARTITION BY s.City ORDER BY o.Total_orders DESC) AS Rank_in_City
    FROM orders o
    JOIN swiggy s ON o.Restaurant_id = s.ID
)
SELECT
    Area,
    City,
    SUM(Total_orders) AS Total_Area_Orders,
    ROUND(AVG(Avg_ratings), 2) AS Average_Rating,
    MAX(Total_orders) AS Max_Area_Orders
FROM RankedOrders
WHERE Rank_in_City <= 5
GROUP BY Area, City
HAVING SUM(Total_orders) > 20000
ORDER BY Total_Area_Orders DESC;
```



Area	City	Total_Area_Orders	Average_Rating	Max_Area_Orders
Chembur	Mumbai	59778	3.18	14975
Piplod	Surat	44648	2.9	14903
Ballygunge	Kolkata	29976	3.35	14996
Kondhwa	Pune	29937	2.9	14980
Central Bangalore	Bangalore	29919	4.2	14960
Ellisbridge	Ahmedabad	29863	3.65	14944
Rohini	Delhi	29820	3.3	14977

Highlights

Insights

- By limiting the analysis to the top 5 restaurants (Rank_in_City <= 5), the query focuses on the best-performing restaurants, eliminating noise from underperforming establishments.

- HAVING SUM(Total_orders) > 20000 filters for areas with significant revenue potential. This highlights regions that contribute meaningfully to overall revenue.

- Areas with a high sum of orders (SUM(Total_orders)) and high ratings are opportunities for further investment (e.g., marketing, logistics improvements).

Use Cases:

- Analyse restaurant data to understand which areas and cities are contributing the most to order volumes.
- Identify low-performing areas or cities (those not meeting the HAVING condition) to target for promotional offers or service improvements.
- Performance Trends: Analyse which cities or areas have the highest average ratings and total orders, helping prioritize delivery expansions or promotional campaigns.
- Areas with high total orders and high average ratings are prime candidates for opening new franchises or expanding existing services.

Conclusion

- The analysis uncovered key trends in restaurant performance, pricing, delivery efficiency, and customer preferences.
- These insights can guide Swiggy in enhancing user experience, optimizing operations, and driving growth.

Future Scope

- Real-Time Data Integration: Incorporate live data for dynamic analysis.
- Customer Feedback: Analyse reviews and ratings to refine service quality.
- Advanced Analytics: Use machine learning to predict demand for dishes, areas, or price categories.
- Scalability: Extend the analysis to other regions or platforms.

Thank You