



ZOMATO SQL ANALYTICS PROJECT



END-TO-END DATA ANALYSIS ON A FOOD DELIVERY MARKETPLACE

A full relational SQL case study analyzing customer behavior, restaurant performance, cuisines, ratings, and order patterns.



PROBLEM STATEMENT

The food delivery platform generates data across multiple systems — restaurants, menu items, orders, customers, cuisines, and ratings.

Right now, this information sits in separate tables, and without proper analysis, it tells the business nothing meaningful.

The core problem for the business is to understand:

- Which restaurants are actually generating revenue

- What food items customers most frequently order

- How demand varies by city and by hour

- Whether ratings influence revenue

- Which customers are loyal

- Which cuisines dominate demand

- Which restaurants are declining in performance

To answer these, the scattered SQL tables must be joined, aggregated, and analyzed to convert raw data into actionable insights.



PROJECT OBJECTIVE

The objective of this project is to build a realistic, analytics-ready SQL database and use it to extract insights that matter to a food-delivery business.

This project aims to:

- Integrate relational tables into a consistent analytical model
- Analyze 1006+ orders to identify revenue leaders and demand patterns
- Explore customer behaviour, repeat orders, and spending distribution
- Study cuisine popularity and city-level variations
- Evaluate restaurant performance using ratings and revenue metrics
- Produce charts and insights that support business decisions
- Deliver an end-to-end SQL case study suitable for portfolio and interviews

This project simulates the analytical responsibilities of a real data analyst at Zomato/Swiggy.



DATA OVERVIEW

█ Restaurants

- 300 restaurants across Indian cities
- Each with name, city, avg_rating, and veg/non-veg indicator

█ Customers

- 500 customers with realistic names, cities, phone numbers

█ Orders

- 400 orders (regenerated with realistic distribution)
- Includes order_time, total_amount, payment_mode, status

█ Order Items

- 1006 order_items linked to orders
- Each with item_id, price, quantity
- Reflecting real mixed-order behaviour

█ Menu Items

- ~904 menu_items
- 2–4 items per restaurant

Veg/non-veg tagging

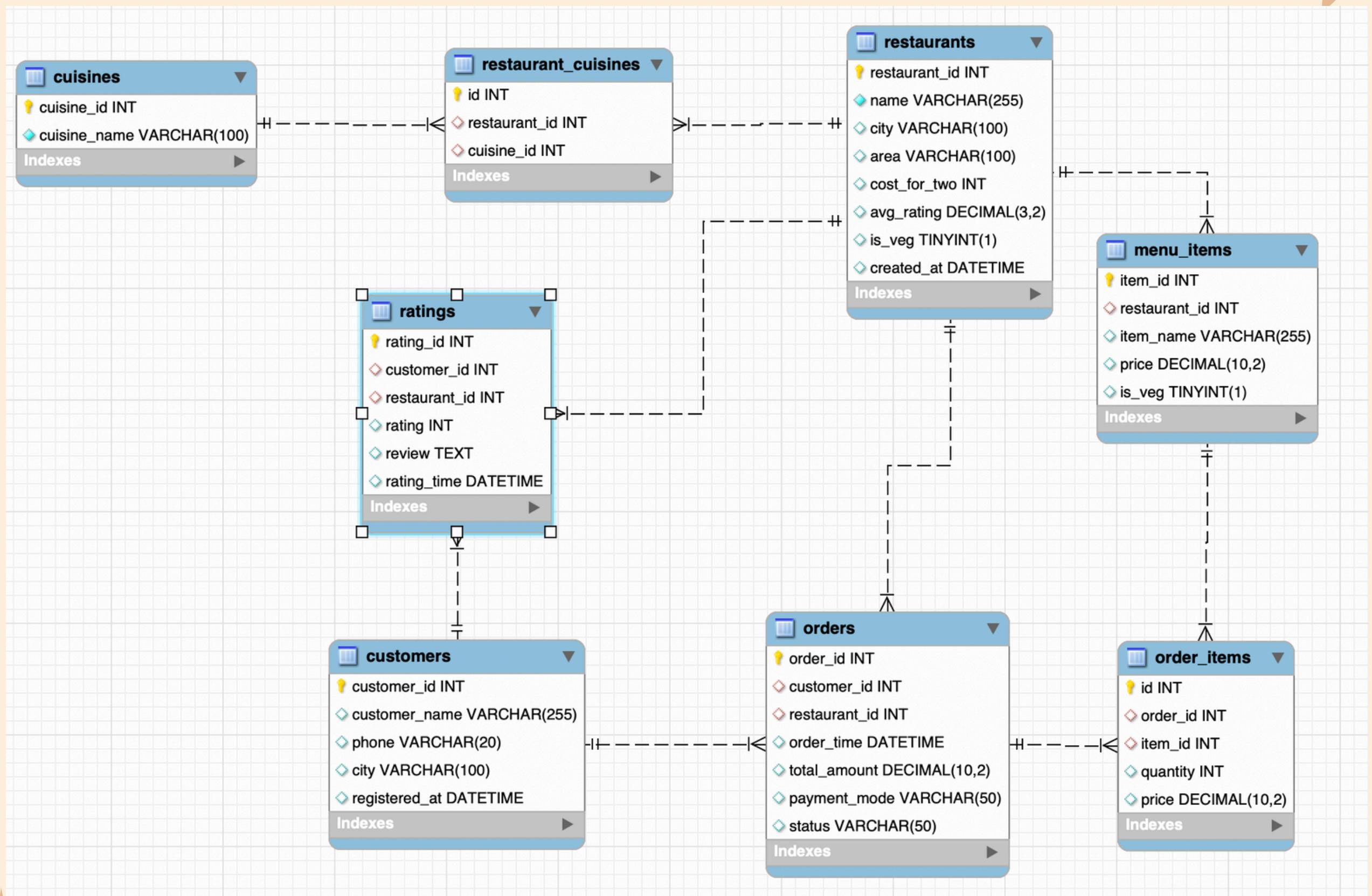
█ Restaurant–Cuisines

- 600 mappings
- Each restaurant mapped to 2 cuisines

█ Ratings

- 350 ratings with review text and timestamps

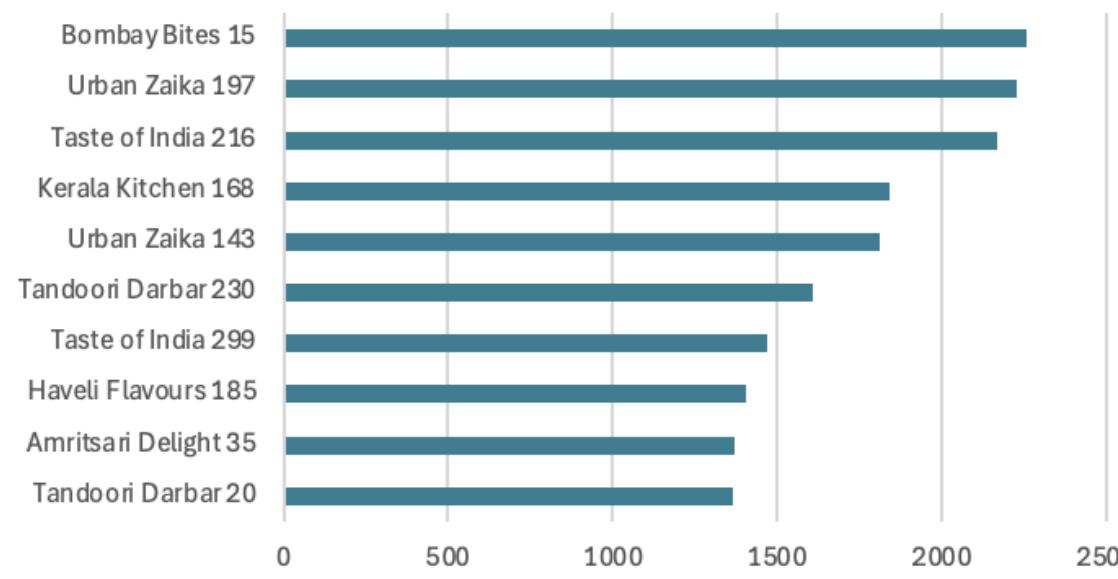
EER Diagram



1. Which restaurants generate the highest revenue?

```
1
2  -- Top 10 revenue-generating restaurants
3 • SELECT
4    r.restaurant_id,
5    r.name AS restaurant_name,
6    r.city,
7    SUM(o.total_amount) AS total_revenue,
8    COUNT(order_id) AS total_orders,
9    ROUND(AVG(r.avg_rating), 2) AS avg_rating  -- AVG() used to comply with GROUP BY; value is already constant per restaurant.
10
11 FROM
12   restaurants AS r
13   JOIN
14     orders AS o ON r.restaurant_id = o.restaurant_id
15 WHERE
16   o.status = 'Delivered' -- only count valid completed orders
17 GROUP BY r.name , r.city , r.restaurant_id
18 ORDER BY total_revenue DESC
19 LIMIT 10;      -- show top 10
20
21
```

Top Revenue Restaurants



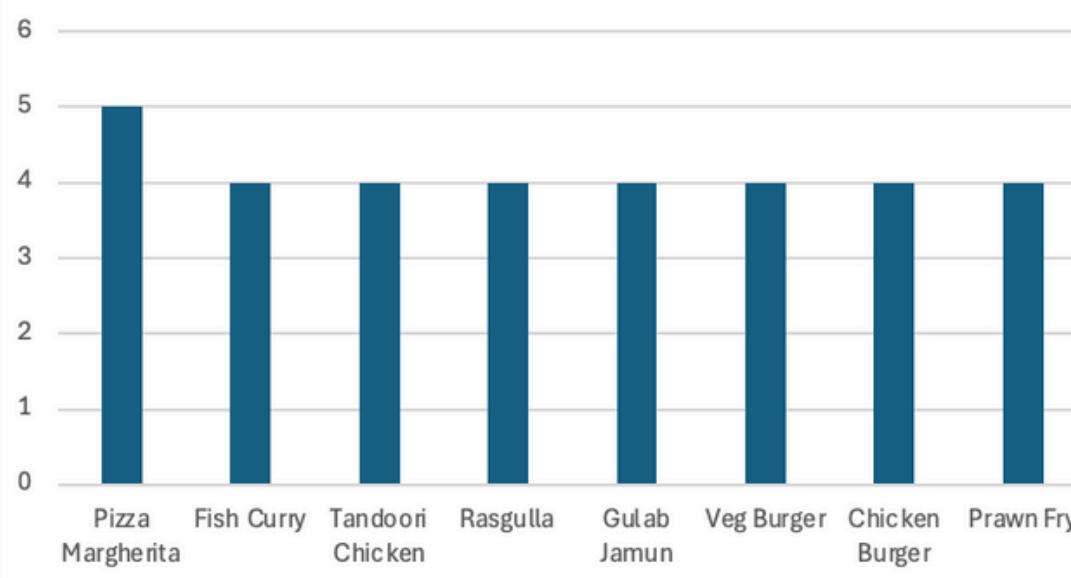
Output-

| restaurant_id | restaurant_name | city | total_revenue | total_orders | avg_rating | |
|---------------|----------------------|-----------|---------------|--------------|------------|--|
| 15 | Bombay Bites 15 | Kolkata | 2259.00 | 2 | 3.54 | |
| 197 | Urban Zaika 197 | Hyderabad | 2227.00 | 2 | 4.57 | |
| 216 | Taste of India 216 | Hyderabad | 2168.00 | 2 | 4.68 | |
| 168 | Kerala Kitchen 168 | Chennai | 1842.00 | 2 | 4.00 | |
| 143 | Urban Zaika 143 | Mumbai | 1810.00 | 2 | 4.72 | |
| 230 | Tandoori Darbar 230 | Delhi | 1608.00 | 2 | 3.99 | |
| 299 | Taste of India 299 | Mumbai | 1469.00 | 2 | 4.78 | |
| 185 | Haveli Flavours 185 | Hyderabad | 1405.00 | 3 | 4.52 | |
| 35 | Amritsari Delight 35 | Pune | 1372.00 | 2 | 4.03 | |
| 20 | Tandoori Darbar 20 | Kolkata | 1367.00 | 3 | 3.95 | |

2. What are the most ordered menu items across the platform?

```
1
2    -- Most frequently ordered menu items across the platform
3
4 • SELECT
5        m.item_name, -- name of menu item
6        COUNT(o.item_id) AS order_count, -- how many time it appeared in order
7        SUM(o.quantity) AS total_quantity -- total item order
8    FROM
9        menu_items AS m
10       JOIN
11        order_items AS o ON m.item_id = o.item_id
12    GROUP BY m.item_id , m.item_name
13    ORDER BY order_count DESC
14    LIMIT 10; -- top 10 order appeared
15
```

Most Ordered Item

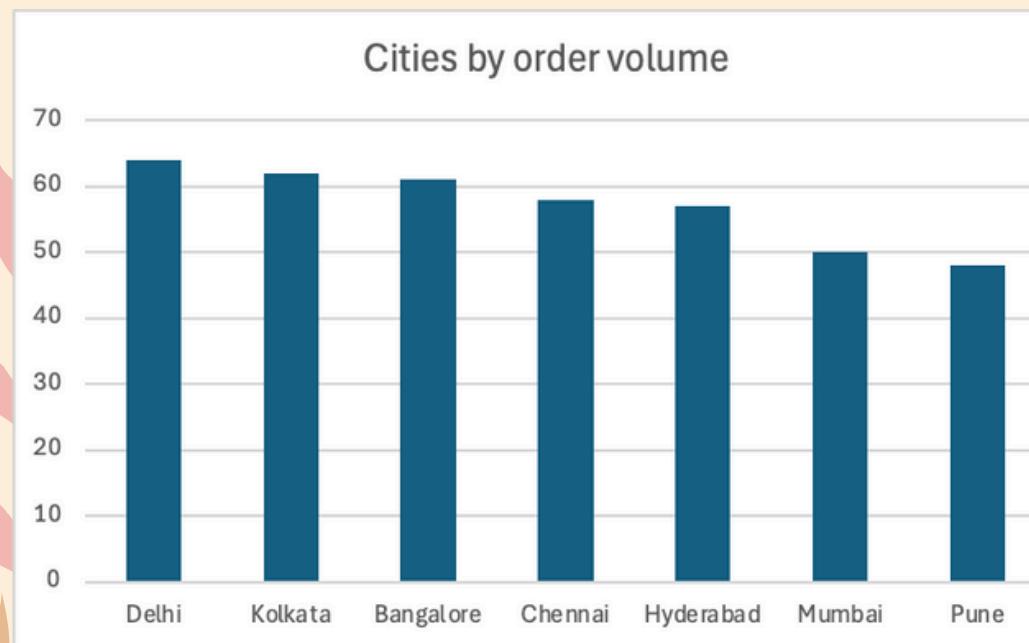


Output-

| item_name | order_count | total_quant... |
|------------------|-------------|----------------|
| Pizza Margherita | 5 | 12 |
| Rasgulla | 4 | 11 |
| Chicken Burger | 4 | 12 |
| Fish Curry | 4 | 11 |
| Gulab Jamun | 4 | 10 |
| Chicken Burger | 4 | 12 |
| Veg Burger | 4 | 10 |
| Prawn Fry | 4 | 11 |
| Tandoori Chicken | 4 | 10 |
| Fish Curry | 4 | 12 |

3. Which cities show the highest order volume?

```
1  -- Order volume by city
2
3 * SELECT
4      r.city,
5      COUNT(order_id) AS total_order -- number of order in cities
6  FROM
7      restaurants AS r
8      JOIN
9      orders AS o ON r.restaurant_id = o.restaurant_id
10     GROUP BY r.city -- each row by city
11     ORDER BY total_order DESC; -- highest ordering city first
12
```

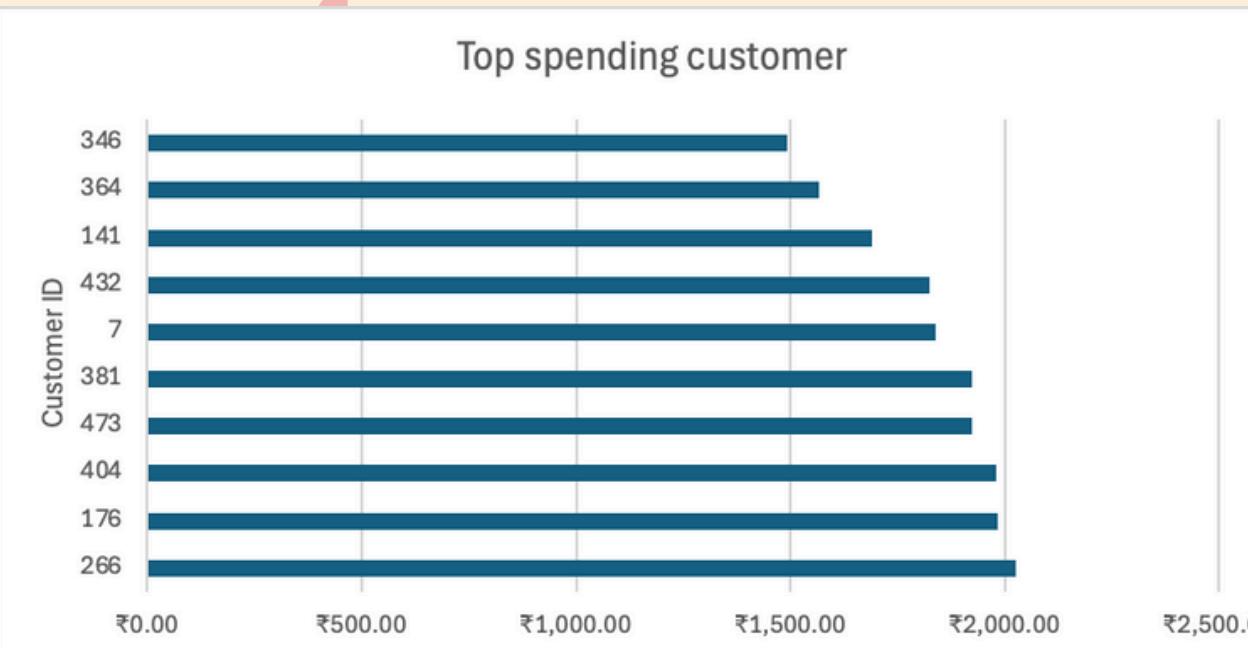


Output--

| city | total_order |
|-----------|-------------|
| Delhi | 64 |
| Kolkata | 62 |
| Bangalore | 61 |
| Chennai | 58 |
| Hyderabad | 57 |
| Mumbai | 50 |
| Pune | 48 |

4. Who are the top 10 highest-spending customers?

```
1      -- Top customers ranked by total spending
2
3
4 •  SELECT
5      c.customer_id,
6      c.customer_name,
7      SUM(o.total_amount) AS total_spend, -- total money spent
8      COUNT(order_id) AS total_order -- number of order placed
9  FROM
10     customers AS c
11     JOIN
12     orders AS o ON c.customer_id = o.customer_id
13  WHERE
14      o.status = 'Delivered' -- only count successful orders
15  GROUP BY c.customer_id , c.customer_name
16  ORDER BY total_spend DESC
17  LIMIT 10;  -- top 10
18
```



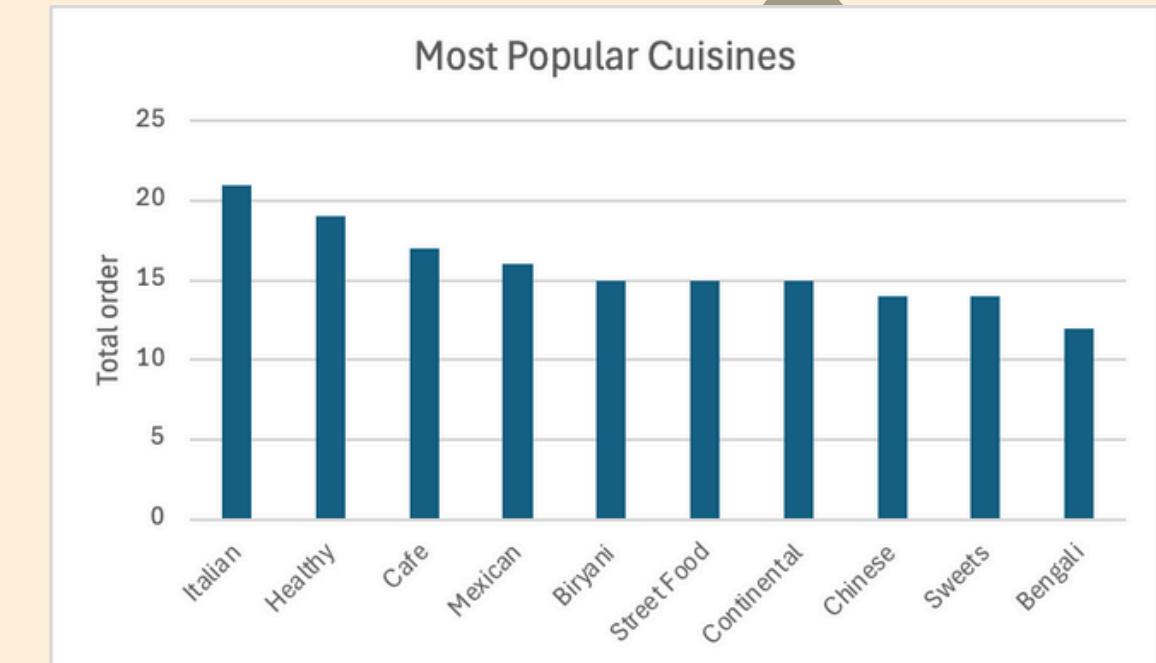
Output-

| customer_id | customer_name | total_spe... | total_order |
|-------------|----------------|--------------|-------------|
| 266 | Aarav Nair | 2027.00 | 2 |
| 176 | Krish Reddy | 1983.00 | 2 |
| 404 | Navya Malhotra | 1981.00 | 2 |
| 473 | Myra Verma | 1923.00 | 2 |
| 381 | Myra Singh | 1922.00 | 2 |
| 7 | Dhruv Verma | 1839.00 | 2 |
| 432 | Vivaan Jha | 1823.00 | 2 |
| 141 | Krish Patel | 1691.00 | 2 |
| 364 | Krish Mehta | 1568.00 | 2 |
| 346 | Diya Bedi | 1493.00 | 2 |

5. What cuisines are most popular based on order volume?

```
1
2      -- Most popular cuisines by order volume
3
4 •  SELECT
5          c.cuisine_name, -- type of cuisine
6          COUNT(order_id) AS total_order
7
8      FROM
9          orders AS o
10         JOIN
11             restaurants AS r ON o.restaurant_id = r.restaurant_id
12         JOIN
13             restaurant_cuisines AS rc ON r.restaurant_id = rc.restaurant_id
14         JOIN
15             cuisines AS c ON rc.cuisine_id = c.cuisine_id
16
17     WHERE
18         o.status = 'Delivered' -- only successful orders
19     GROUP BY c.cuisine_name
20     ORDER BY total_order DESC
21     LIMIT 10;    -- top 10
```

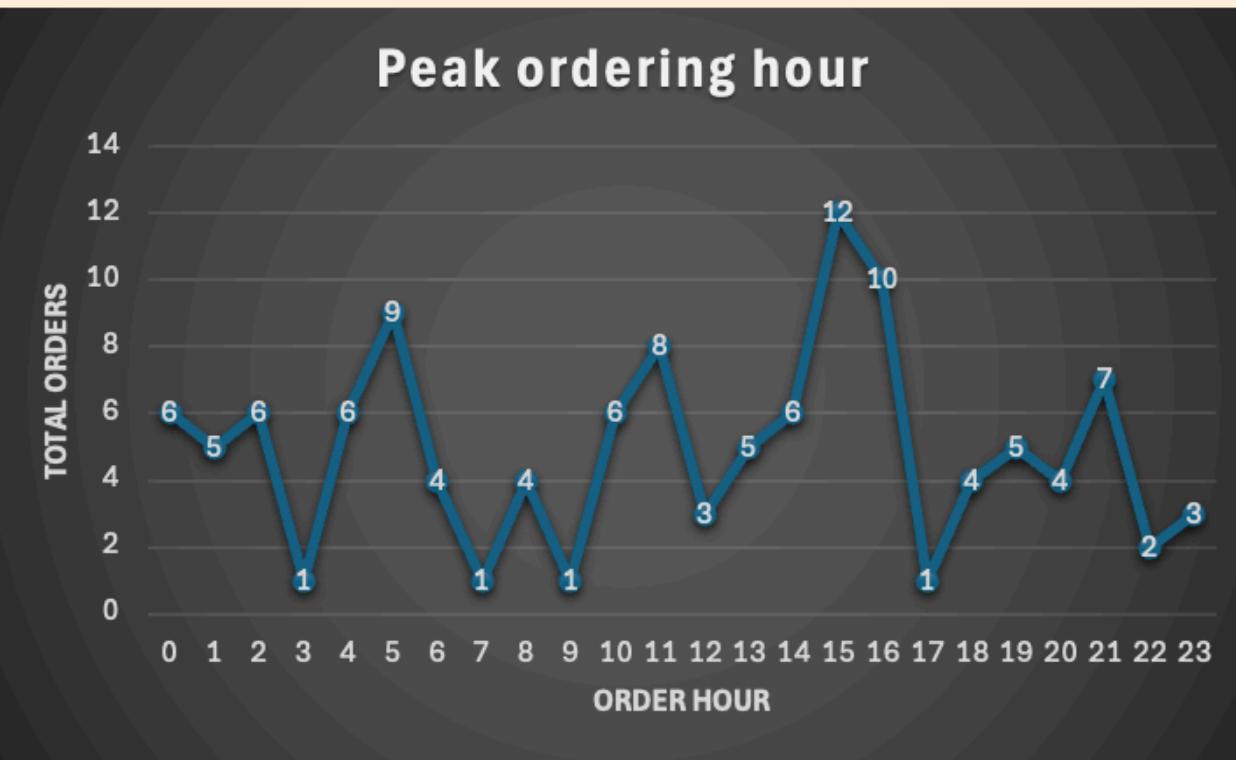
Output-



| cuisine_name | total_order |
|--------------|-------------|
| Italian | 21 |
| Healthy | 19 |
| Cafe | 17 |
| Mexican | 16 |
| Continental | 15 |
| Street Food | 15 |
| Biryani | 15 |
| Sweets | 14 |
| Chinese | 14 |
| Bengali | 12 |

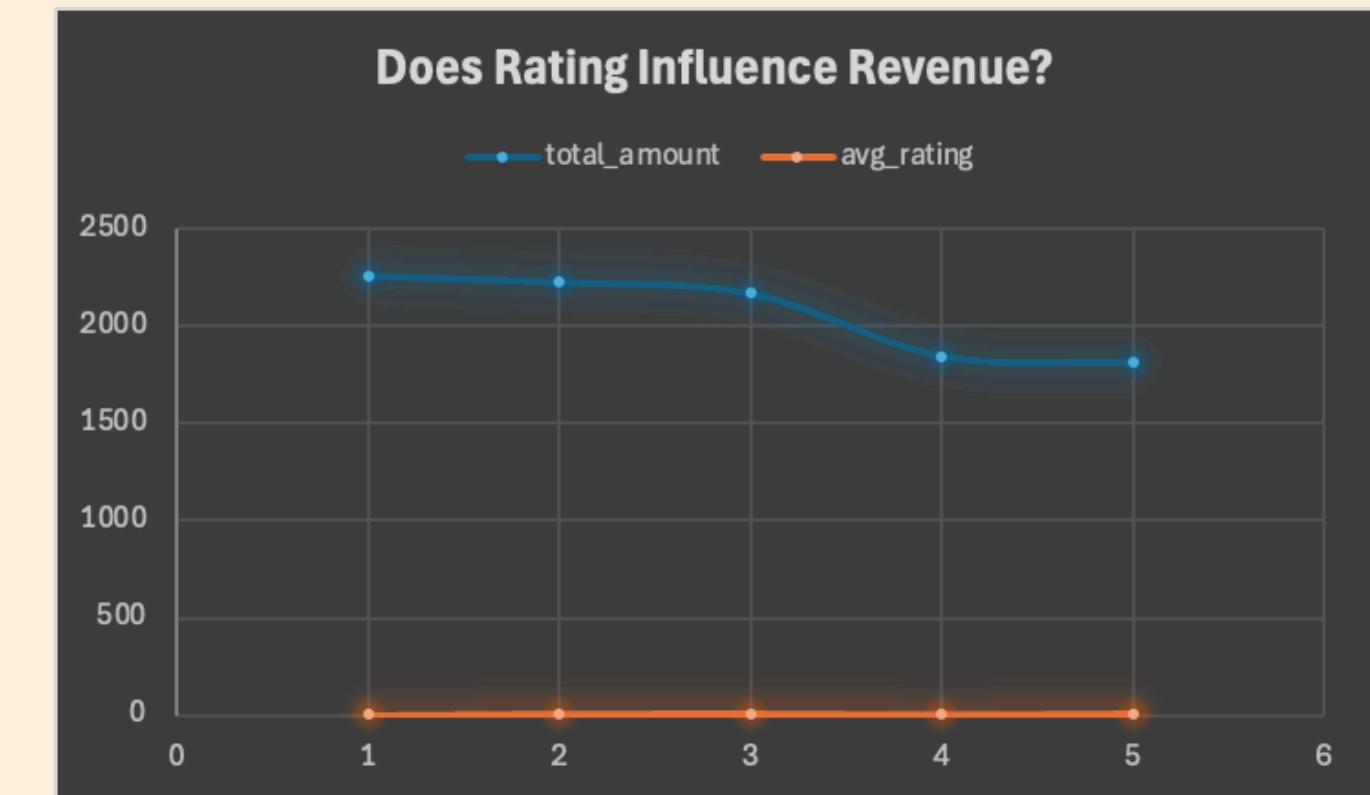
6. What hours of the day see the highest number of orders?

```
1
2    -- Peak ordering hours across the platform
3
4 • SELECT
5        HOUR(order_time) AS order_hour, -- hour of the day (0-23)
6        COUNT(order_id) AS total_orders
7 FROM
8     orders
9 WHERE
10       status = 'Delivered' -- only include completed orders
11 GROUP BY order_hour
12 ORDER BY total_orders DESC;
13
```



7. Do high-rated restaurants earn more revenue than low-rated ones?

```
2  -- Does a higher rating correlate with higher revenue?  
3  
4 • SELECT  
5   r.name, -- restaurant name  
6   CASE  
7     WHEN avg_rating >= 4.5 THEN 'Excellent (4.5 - 5.0)'  
8     WHEN avg_rating >= 4.0 THEN 'Good (4.0 - 4.4)'  
9     WHEN avg_rating >= 3.5 THEN 'Average (3.5 - 3.9)'  
10    ELSE 'Low (< 3.5)'  
11  END rating_band,          -- rating groups  
12  SUM(o.total_amount) AS total_amount,  
13  ROUND(AVG(r.avg_rating), 2) AS avg_rating  
14 FROM  
15   restaurants AS r  
16   JOIN  
17     orders AS o ON r.restaurant_id = o.restaurant_id  
18 WHERE  
19   o.status = 'Delivered'  -- only successful orders  
20 GROUP BY r.name , rating_band  
21 ORDER BY total_amount DESC  
22 LIMIT 5;                -- top 5
```



Output-

| name | rating_band | total_amou... | avg_rating |
|--------------------|-----------------------|---------------|------------|
| Bombay Bites 15 | Average (3.5 - 3.9) | 2259.00 | 3.54 |
| Urban Zaika 197 | Excellent (4.5 - 5.0) | 2227.00 | 4.57 |
| Taste of India 216 | Excellent (4.5 - 5.0) | 2168.00 | 4.68 |
| Kerala Kitchen 168 | Good (4.0 - 4.4) | 1842.00 | 4.00 |
| Urban Zaika 143 | Excellent (4.5 - 5.0) | 1810.00 | 4.72 |

8. Which restaurants have declining ratings over time?

```
1      -- Identify restaurants whose ratings drop over time
2 • WITH rating_trend AS (
3     SELECT
4         rt.restaurant_id,
5             r.name AS restaurant_name,
6             rt.rating,
7             rt.rating_time,
8
9             LAG(rt.rating) OVER (
10                 PARTITION BY rt.restaurant_id
11                 ORDER BY rt.rating_time
12             ) AS prev_rating
13     FROM ratings rt
14     JOIN restaurants r
15         ON rt.restaurant_id = r.restaurant_id
16 )
17
18     SELECT
19         restaurant_id,
20         restaurant_name,
21         COUNT(*) AS decline_count    -- how many times the rating decreased
22     FROM rating_trend
23     WHERE prev_rating IS NOT NULL    -- ignore the first rating of each restaurant
24         AND rating < prev_rating    -- drop detected
25     GROUP BY restaurant_id, restaurant_name
26     ORDER BY decline_count DESC;    -- worst performers first
27
28
```



| restaurant_id | restaurant_name | decline_cou... |
|---------------|-------------------------|----------------|
| 26 | Amritsari Delight 26 | 2 |
| 265 | Haveli Flavours 265 | 2 |
| 180 | Southern Spice 180 | 2 |
| 59 | Kerala Kitchen 59 | 2 |
| 4 | Taste of India 4 | 1 |
| 295 | Amritsari Delight 295 | 1 |
| 25 | Kerala Kitchen 25 | 1 |
| 21 | Nawabi Dastarkhwan 21 | 1 |
| 30 | Kerala Kitchen 30 | 1 |
| 47 | Curry Point 47 | 1 |
| 20 | Tandoori Darbar 20 | 1 |
| 72 | Punjabi Rasoi 72 | 1 |
| 75 | Amritsari Delight 75 | 1 |
| 88 | Nawabi Dastarkhwan 88 | 1 |
| 92 | Haveli Flavours 92 | 1 |
| 103 | Haveli Flavours 103 | 1 |
| 109 | Kerala Kitchen 109 | 1 |
| 112 | Biryani House 112 | 1 |
| 119 | Masala Square 119 | 1 |
| 121 | Kerala Kitchen 121 | 1 |
| 123 | Masala Square 123 | 1 |
| 125 | Southern Spice 125 | 1 |
| 126 | Taste of India 126 | 1 |
| 135 | Kerala Kitchen 135 | 1 |
| 139 | Kerala Kitchen 139 | 1 |
| 141 | Urban Zaika 141 | 1 |
| 143 | Urban Zaika 143 | 1 |
| 151 | Masala Square 151 | 1 |
| 154 | Bombay Bites 154 | 1 |
| 165 | Royal Kitchen 165 | 1 |
| 172 | Tandoori Darbar 172 | 1 |
| 18 | Royal Kitchen 18 | 1 |
| 183 | Southern Spice 183 | 1 |
| 190 | Tandoori Darbar 190 | 1 |
| 204 | Nawabi Dastarkhwan 2... | 1 |
| 209 | Spice Junction 209 | 1 |
| 210 | Nawabi Dastarkhwan 2... | 1 |
| 213 | Haveli Flavours 213 | 1 |
| 218 | Bombay Bites 218 | 1 |
| 226 | Royal Kitchen 226 | 1 |
| 236 | Tandoori Darbar 236 | 1 |
| 253 | Biryani House 253 | 1 |
| 13 | Tandoori Darbar 13 | 1 |
| 266 | Spice Junction 266 | 1 |

Output-

9. Which customers show repeat behaviour (loyal customers)?

```
1
2      -- Identify loyal customers based on repeat orders
3
4 •  SELECT
5      c.customer_id,
6      c.customer_name,
7      COUNT(o.order_id) AS total_order -- number of orders delivered
8  FROM
9      customers AS c
10     JOIN
11         orders AS o ON c.customer_id = o.customer_id
12  WHERE
13      o.status = 'Delivered'
14  GROUP BY c.customer_id , c.customer_name
15  HAVING COUNT(o.order_id) >= 2 -- threshold for loyalty
16  ORDER BY total_order DESC; -- most loyal customer first
17
```

Output-

| customer_id | customer_name | total_order |
|-------------|----------------|-------------|
| 346 | Diya Bedi | 2 |
| 7 | Dhruv Verma | 2 |
| 478 | Aarya Rathod | 2 |
| 266 | Aarav Nair | 2 |
| 141 | Krish Patel | 2 |
| 387 | Anaya Jha | 2 |
| 375 | Riya Rathod | 2 |
| 404 | Navya Malhotra | 2 |
| 364 | Krish Mehta | 2 |
| 176 | Krish Reddy | 2 |
| 491 | Arjun Verma | 2 |
| 381 | Myra Singh | 2 |
| 457 | Anaya Desai | 2 |
| 473 | Myra Verma | 2 |
| 432 | Vivaan Jha | 2 |

10. Which restaurants have the highest average order value (AOV)?

```
1
2      -- Restaurants ranked by highest Average Order Value (AOV)
3
4 •  SELECT
5
6      r.name AS restaurant_name,
7      r.city,
8      SUM(o.total_amount) AS total_revenue,
9      COUNT(o.order_id) AS total_orders,
10     ROUND(SUM(o.total_amount) / COUNT(o.order_id),
11           2) AS avg_order_value -- AOV
12
13    FROM
14        restaurants AS r
15        JOIN
16            orders AS o ON r.restaurant_id = o.restaurant_id
17
18    WHERE
19        o.status = 'Delivered'
20    GROUP BY r.name , r.city
21    HAVING total_orders > 0
22    ORDER BY avg_order_value DESC;
```

Output-

| restaurant_name | city | total_revenue | total_orders | avg_order_value |
|-----------------------|-----------|---------------|--------------|-----------------|
| Urban Zaika 39 | Kolkata | 1192.00 | 1 | 1192.00 |
| Curry Point 158 | Kolkata | 1187.00 | 1 | 1187.00 |
| Amritsari Delight 120 | Kolkata | 1186.00 | 1 | 1186.00 |
| Royal Kitchen 270 | Hyderabad | 1176.00 | 1 | 1176.00 |
| Spice Junction 16 | Chennai | 1170.00 | 1 | 1170.00 |
| Amritsari Delight 176 | Kolkata | 1163.00 | 1 | 1163.00 |
| Biryani House 53 | Mumbai | 1155.00 | 1 | 1155.00 |
| Bombay Bites 15 | Kolkata | 2259.00 | 2 | 1129.50 |
| Taste of India 8 | Bangalore | 1129.00 | 1 | 1129.00 |
| Tandoori Darbar 169 | Delhi | 1119.00 | 1 | 1119.00 |
| Amritsari Delight 298 | Delhi | 1116.00 | 1 | 1116.00 |
| Urban Zaika 197 | Hyderabad | 2227.00 | 2 | 1113.50 |
| Kerala Kitchen 107 | Kolkata | 1102.00 | 1 | 1102.00 |
| Punjabi Rasoi 56 | Chennai | 1102.00 | 1 | 1102.00 |
| Taste of India 192 | Chennai | 1100.00 | 1 | 1100.00 |
| Taste of India 216 | Hyderabad | 2168.00 | 2 | 1084.00 |
| Haveli Flavours 178 | Pune | 1073.00 | 1 | 1073.00 |
| Bombay Bites 296 | Hyderabad | 1070.00 | 1 | 1070.00 |
| Masala Square 150 | Pune | 1054.00 | 1 | 1054.00 |
| Tandoori Darbar 162 | Bangalore | 1051.00 | 1 | 1051.00 |
| Tandoori Darbar 80 | Pune | 1048.00 | 1 | 1048.00 |
| Biryani House 244 | Bangalore | 1041.00 | 1 | 1041.00 |
| Spice Junction 28 | Bangalore | 1038.00 | 1 | 1038.00 |
| Southern Spice 238 | Bangalore | 1013.00 | 1 | 1013.00 |
| Nawabi Dastarkhw... | Mumbai | 1013.00 | 1 | 1013.00 |
| Southern Spice 258 | Bangalore | 950.00 | 1 | 950.00 |
| Amritsari Delight 26 | Bangalore | 944.00 | 1 | 944.00 |
| Masala Square 62 | Pune | 943.00 | 1 | 943.00 |
| Bombay Bites 34 | Delhi | 932.00 | 1 | 932.00 |
| Haveli Flavours 7 | Pune | 927.00 | 1 | 927.00 |
| Kerala Kitchen 168 | Chennai | 1842.00 | 2 | 921.00 |
| Urban Zaika 143 | Mumbai | 1810.00 | 2 | 905.00 |
| Urban Zaika 248 | Chennai | 896.00 | 1 | 896.00 |
| Southern Spice 40 | Chennai | 888.00 | 1 | 888.00 |
| Bombay Bites 292 | Hyderabad | 883.00 | 1 | 883.00 |
| Spice Junction 51 | Hyderabad | 868.00 | 1 | 868.00 |
| Taste of India 4 | Delhi | 834.00 | 1 | 834.00 |
| Tandoori Darbar 117 | Bangalore | 818.00 | 1 | 818.00 |
| Punjabi Rasoi 193 | Delhi | 816.00 | 1 | 816.00 |
| Tandoori Darbar 230 | Delhi | 1608.00 | 2 | 804.00 |
| Tandoori Darbar 87 | Mumbai | 802.00 | 1 | 802.00 |
| Curry Point 63 | Kolkata | 760.00 | 1 | 760.00 |
| Royal Kitchen 165 | Kolkata | 753.00 | 1 | 753.00 |
| Punjabi Rasoi 89 | Hyderabad | 752.00 | 1 | 752.00 |
| Royal Kitchen 84 | Mumbai | 736.00 | 1 | 736.00 |



FINAL RECOMMENDATIONS

1. Strengthen Partnerships With High-Performing Restaurants

- Top revenue restaurants maintain strong order flow despite the small dataset.
- These restaurants should be prioritized for promotional partnerships, visibility boosts, and featured listings.

2. Expand Supply for High-Demand Cuisines

- Certain cuisines consistently dominate delivered orders (based on your Q5).
- Onboard more restaurants offering these cuisines to reduce competition load and improve availability.

3. Improve Performance of Low-Rated but High-Volume Restaurants

- Some restaurants show high order volume but weak or declining ratings (Q7 & Q8).
- Recommend quality audits, faster delivery coordination, and menu improvement strategies.

4. Optimize Delivery Fleet for Peak Ordering Hours

- Ordering peaks within specific hours (Q6).
- Allocate the highest delivery capacity between those hours to reduce delays and cancellations.

5. Build Loyalty Programs for Repeat Customers

- A segment of customers shows repeat purchase behavior (Q9).
- Personalized offers, loyalty points, or re-engagement campaigns can increase their lifetime value.

6. Promote High AOV Restaurants Strategically

- High AOV restaurants contribute significant revenue per order (Q10).
- Recommend targeted premium campaigns or bundled offers to maintain customer conversion.

CONCLUSION

This SQL project reconstructed a complete food-delivery ecosystem using eight interconnected tables, reflecting real operational entities such as restaurants, customers, orders, menu items, cuisines, and ratings. Through structured querying, KPI extraction, and visual analysis, the project uncovered meaningful insights into restaurant performance, customer behavior, cuisine preferences, rating dynamics, and order-time patterns.

The analysis reveals:

- Clear revenue and order leaders among restaurants
- Strong demand patterns centered around specific cuisines
- Predictable peak-hour ordering behavior
- A relationship between restaurant ratings and revenue
- Distinct segments of loyal, high-value customers
- AOV variation helping identify premium vs. volume-driven restaurants

These insights lead to actionable recommendations for strengthening restaurant partnerships, improving customer retention, optimizing delivery operations, and enhancing overall marketplace performance.

The project demonstrates the full workflow of a data analyst – from data modeling and SQL querying to insight generation and business-oriented interpretation – making it a strong showcase of analytical and technical skill.