

Zomato SQL Analytics Project



Turning SQL Queries into Actionable Business Insights



HARSH SONI

Data Analytics

MySQL

Comprehensive analysis of customer behavior, revenue trends, and operational efficiency



Project Overview

◎ What This Project Is About

A comprehensive SQL analytics project using the Zomato dataset to analyze customer behavior, revenue trends, restaurant performance, rider efficiency, churn patterns, and operational insights using advanced SQL techniques.

Dataset Description

- Customers:** User profiles, demographics, and ordering history
- Orders:** Transaction details, order items, amounts, timestamps
- Restaurants:** Restaurant info, locations, cuisines
- Payments:** Payment methods and transaction status
- Deliveries:** Delivery status, rider assignments, timestamps
- Riders:** Delivery partner profiles and performance metrics

💡 Business Value

- Customer Retention:** Identify churn patterns and high-value customers for targeted retention strategies
- Revenue Optimization:** Analyze spending patterns and AOV to maximize customer lifetime value
- Operational Efficiency:** Evaluate rider performance and delivery times to improve service quality
- Restaurant Partnerships:** Identify top-performing restaurants and growth opportunities by city

</>SQL Techniques Used

- CTEs
- Window Functions
- JOINS
- Subqueries
- CASE Statements
- Aggregations



Project Objectives



Customer Analytics

Analyze customer behavior patterns, identify high-value segments, and understand ordering preferences to drive personalized marketing and retention strategies.



Revenue Insights

Evaluate revenue trends, average order values, and customer lifetime value to optimize pricing strategies and maximize platform profitability.



Restaurant Performance

Assess restaurant performance metrics, identify top performers by city, and analyze growth trends to strengthen partnerships and optimize marketplace quality.



Operational Efficiency

Measure delivery performance, rider efficiency, and order fulfillment rates to enhance operational excellence and customer satisfaction.



Churn Detection

Identify at-risk customers and restaurants with high cancellation rates to implement proactive retention and quality improvement measures.

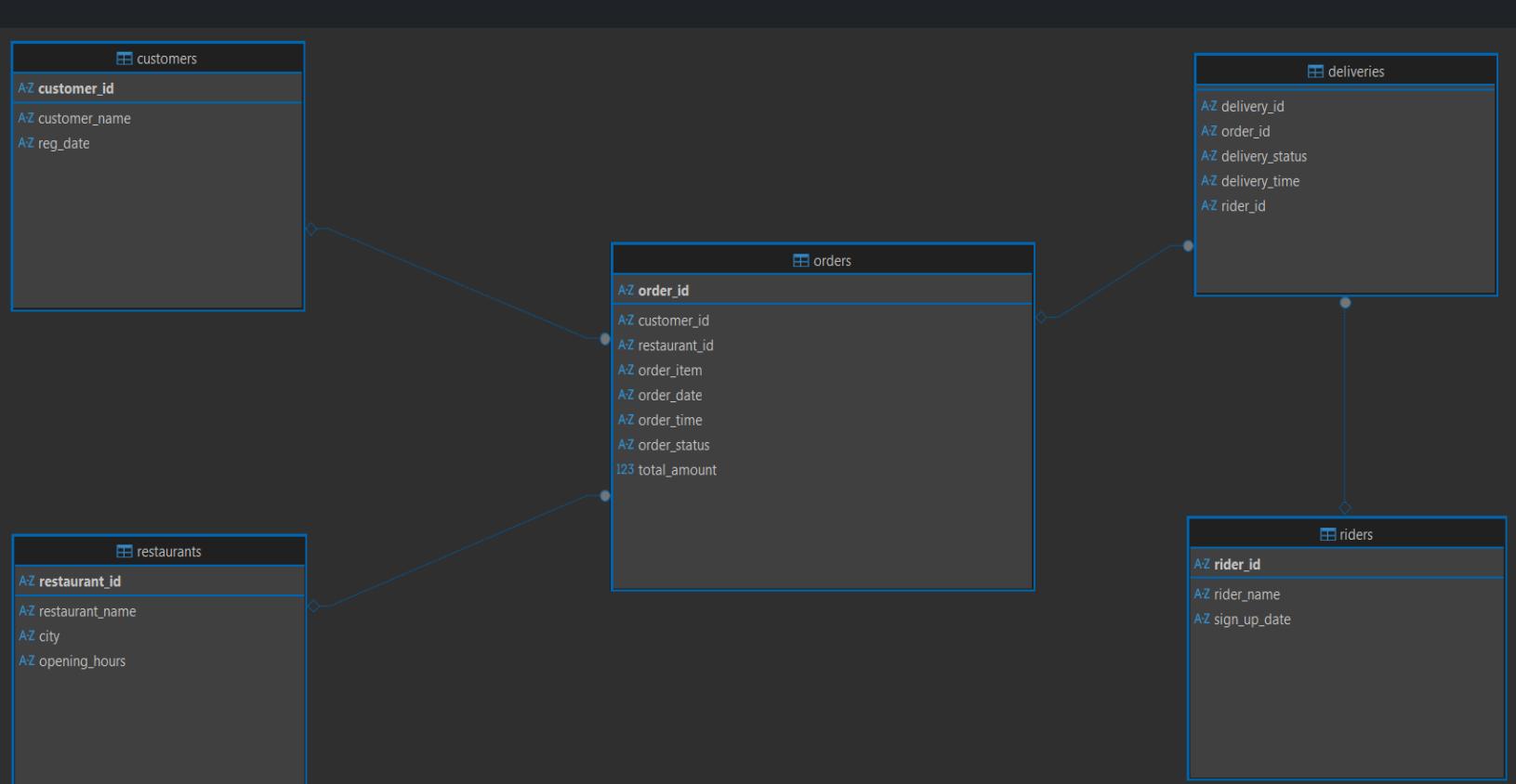


Advanced SQL Skills

Demonstrate proficiency in complex SQL techniques including CTEs, window functions, and advanced aggregations for real-world business problem-solving.



Database ER Diagram



Tables

- customers
- orders
- restaurants
- deliveries
- riders

Primary Keys

- customers.customer_id
- orders.order_id
- restaurants.restaurant_id
- deliveries.delivery_id
- riders.rider_id

Foreign Keys

- orders.customer_id → customers
- orders.restaurant_id → restaurants
- deliveries.order_id → orders
- deliveries.rider_id → riders

Customer Favorite Dishes

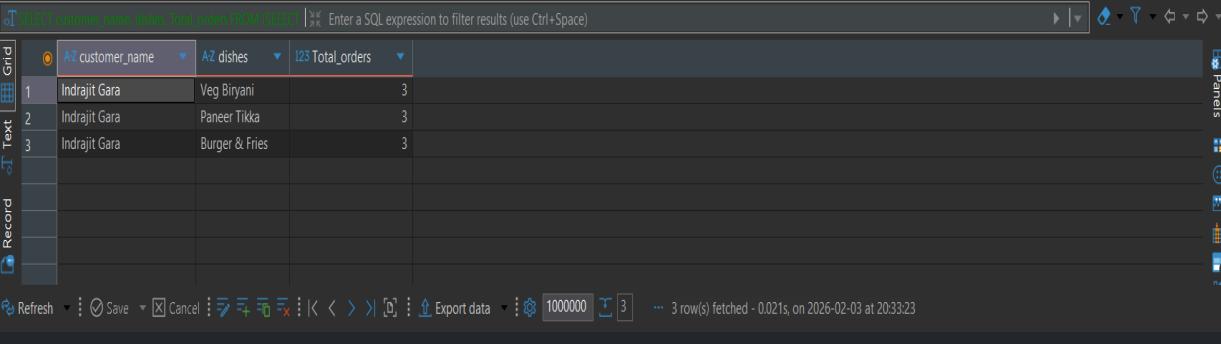
Business Question

Determine the top three most frequently ordered dishes by the customer **Indrajit Gara** over the last 12 months, based on total order frequency.

SQL Query

```
SELECT customer_name,
       dishes,
       Total_orders
  FROM (
    SELECT c.customer_id,
           c.customer_name,
           o.order_item_dishes,
           Count(*) Total_orders,
           DENSE_RANK() OVER(ORDER BY Count(*)) as Ranks
      FROM orders o
     JOIN customers c
       ON o.customer_id = c.customer_id
     WHERE o.order_date >= CURRENT_DATE - INTERVAL 1 YEAR
     AND
       c.customer_name = "Indrajit Gara"
   GROUP BY 1,2,3
  ORDER BY 4 DESC) as t
 WHERE Ranks <= 3
 limit 3;
```

Output Summary



A screenshot of a database interface showing a query results grid. The query is:

```
SELECT customer_name, dishes, Total_orders FROM (
```

The results grid has columns: Grid, Text, Record. The data shows three rows for customer 'Indrajit Gara' with dishes 'Veg Biryani', 'Paneer Tikka', and 'Burger & Fries', all having a 'Total_orders' count of 3.

Grid	customer_name	dishes	Total_orders
1	Indrajit Gara	Veg Biryani	3
2	Indrajit Gara	Paneer Tikka	3
3	Indrajit Gara	Burger & Fries	3

At the bottom, there are buttons for Refresh, Save, Cancel, and Export data, along with a page number '1000000' and a '3' indicating 3 rows fetched.

Key Insight

The customer Indrajit Gara shows a clear preference for a small set of dishes, with multiple items having the same highest order frequency. This indicates consistent repeat ordering behavior, suggesting strong customer loyalty toward specific menu items.

Business Recommendation

- ✓ Bundle these top dishes for combo offers
- ✓ Offer personalized discounts to increase retention
- ✓ Prioritize in recommendation systems

Popular Time Slots

Business Question

Analyze order volume distribution across **2-hour time slots** to identify peak ordering periods on the Zomato platform.

SQL Query

```
SELECT
    FLOOR(EXTRACT(HOUR FROM order_time)/2)*2 AS Start_Time,
    FLOOR(EXTRACT(HOUR FROM order_time)/2)*2 + 2 AS End_Time,
    COUNT(*) AS Total_Order
FROM orders
GROUP BY 1,2
ORDER BY 3 DESC;
```

Output Summary

	Start_Time	End_Time	Total_Order
1	14	16	16,949
2	20	22	16,760
3	12	14	16,613
4	10	12	16,579
5	18	20	16,568
6	16	18	16,531

Key Insight

Order volume peaks during the 14:00–16:00 and 20:00–22:00 time slots, indicating strong demand during lunch and late-evening hours on the Zomato platform.

Business Recommendation

- ✓ These peak windows can be leveraged for dynamic pricing, flash offers, and restaurant promotions.
- ✓ Zomato can optimize delivery partner allocation and ETA predictions during high-demand periods to improve customer experience.

High-Frequency Customer AOV

Business Question

Evaluate the **Average Order Value (AOV)** of customers on the Zomato platform who have placed more than 20 orders, and rank these high-frequency customers based on their AOV.

SQL Query

```
SELECT
    C.customer_name,
    ROUND(avg(o.total_amount),2) aov
FROM orders o
JOIN customers c
on o.customer_id = c.customer_id
GROUP BY 1
HAVING COUNT(order_id) > 20
ORDER BY 2 DESC;
```

Output Summary

	AZ customer_name	123 aov
1	Ekiya Sarin	253.24
2	Gagan Dutta	245.02
3	Krishna Dass	244.83
4	Garima Koshy	243.72
5	Pushti Kakar	243.53
6	Sudiksha Radhakrishnan	243.21
7	Harini Chokshi	241.14
8	Bachittar Ray	240.18

Key Insight

Customers with more than 20 orders demonstrate consistently higher Average Order Value, indicating that repeat customers contribute disproportionately to overall revenue on Zomato.

Business Recommendation

- ✓ Such users can be tagged as High-Value Customers (HVCs)..
- ✓ Zomato can design personalized recommendations, premium memberships (Zomato Gold), and upsell strategies to further increase Customer Lifetime Value (CLV).

High-Value Customers

Business Question

Identify high-value customers on the Zomato platform who have spent more than ₹5,000 on food orders, and rank them based on their total lifetime spend.

</>SQL Query

```
SELECT
c.customer_name,
c.customer_id,
ROUND(SUM(o.total_amount),0) AS total_amount
FROM customers c
JOIN orders o
ON c.customer_id = o.customer_id
GROUP BY
c.customer_name,
c.customer_id
HAVING SUM(o.total_amount) > 5000
ORDER BY total_amount DESC
LIMIT 5;
```

Output Summary

	A-Z customer_name	A-Z customer_id	I23 total_amount
1	Ekiya Sarin	C5754	5,571
2	Dev Banerjee	C8162	5,173
3	Jagat Khosla	C2197	5,163
4	Samaksh Krishnan	C8116	5,131

Key Insight

A small group of customers contributes significantly to total revenue, with each spending over ₹5,000 on food orders . This highlights the presence of high-value users who drive disproportionate revenue on the Zomato platform.

 Business Recommendation

- ✓ Ideal targets for Zomato Gold, exclusive coupons, and priority delivery
 - ✓ Helps Zomato focus on CLV (Customer Lifetime Value) rather than only order volume

Non-Delivered Orders Analysis

Business Question

Identify restaurants on the Zomato platform with orders that were placed but **not delivered**, and analyze the volume of non-delivered orders at the restaurant level.

SQL Query

```
SELECT  
    r.restaurant_name,  
    COUNT(o.order_id) AS cnt_not_delivered_orders  
FROM  
    orders o  
LEFT JOIN  
    restaurants r ON o.restaurant_id = r.restaurant_id  
LEFT JOIN  
    deliveries d ON o.order_id = d.order_id  
WHERE  
    d.delivery_id IS NULL  
GROUP BY 1  
ORDER BY 2 DESC;
```

Output Summary

	AZ restaurant_name	123 cnt_not_delivered_orders
1	Nehas Bistro	220
2	Abhas Dhaba	177
3	Jagdishs Point	118
4	Madhavs Grill	117
5	Anyas Bistro	116
6	Wardas Cafe	112
7	Warinders Grill	112
8	Samuels Resto	107

Key Insight

Certain restaurants show a significantly higher number of non-delivered orders, indicating potential operational bottlenecks, delivery partner shortages, or restaurant-level inefficiencies.

Business Recommendation

- ✓ Zomato can optimize delivery partner allocation, ETA accuracy, and restaurant onboarding quality checks.
- ✓ Reducing non-delivered orders directly improves customer trust, platform ratings, and repeat usage.

Top Revenue Restaurants by City

Business Question

Rank restaurants on the Zomato platform by their **total revenue** generated over the **past year**, and identify the **top-revenue-generating** restaurant within each **city**.

SQL Query

```

SELECT
    city,
    restaurant_id,
    restaurant_name,
    Total_Rev
FROM (
    SELECT
        r.city,
        r.restaurant_id,
        r.restaurant_name,
        ROUND(SUM(o.total_amount), 0) AS Total_Rev,
        RANK() OVER (
            PARTITION BY r.city
            ORDER BY ROUND(SUM(o.total_amount), 0) DESC
        ) AS Ranks
    FROM orders o
    JOIN restaurants r
    ON o.restaurant_id = r.restaurant_id
    WHERE o.order_date >= CURRENT_DATE - INTERVAL 1 YEAR
    GROUP BY r.city, r.restaurant_id, r.restaurant_name
) Restaurant_Revenue_Ranking
WHERE Ranks = 1
ORDER BY city;

```

Output Summary

	AZ city	AZ restaurant_id	AZ restaurant_name	123 Total_Rev
1	Ahmedabad	R72	Poojas Grill	158,749
2	Bengaluru	R18	Azads Resto	191,277
3	Bhopal	R58	Iryas Bistro	156,306
4	Chennai	R22	Jatins Dhaba	179,758
5	Delhi	R5	Nehas Bistro	184,822
6	Hyderabad	R24	Hredhaans Point	206,877
7	Indore	R53	Xitis Cafe	163,935
8	Jaipur	R64	Vedhikas Resto	168,347
9	Kolkata	R33	Timothys Kitchen	193,745
10	Lucknow	R79	Iras Point	146,390

Key Insight

Each city has a distinct top-revenue-generating restaurant, indicating that revenue leadership is highly localized rather than dominated by a single national player.

Business Recommendation

- ✓ Zomato can feature top-earning restaurants in each city for promotions and visibility boosts
- ✓ Useful for commission optimization, premium placement, and partnership strategy.

Popular Dishes by City

Business Question

Identify the most popular dish in each city on the Zomato platform based on total order volume, to understand regional food preferences.

SQL Query

```
SELECT city, order_item, Total_Orders
FROM (
  SELECT
    r.city,
    o.order_item,
    COUNT(o.order_id) AS Total_Orders,
    RANK() OVER (
      PARTITION BY r.city
      ORDER BY COUNT(o.order_id) DESC
    ) AS Rank_by_city
  FROM restaurants r
  JOIN orders o
  ON r.restaurant_id = o.restaurant_id
  GROUP BY r.city, o.order_item
) ranked_data
WHERE Rank_by_city = 1
ORDER BY city;
```

Output Summary

	AZ city	AZ order_item	123 Total_Orders
1	Ahmedabad	Tandoori Roti	699
2	Bengaluru	Burger & Fries	447
3	Bhopal	Pizza Margherita	519
4	Chennai	Tandoori Roti	422
5	Delhi	Veg Biryani	424
6	Hyderabad	Butter Chicken	510
7	Indore	Fried Rice	666
8	Jaipur	Dal Makhani	536
9	Kolkata	Veg Biryani	681

Key Insight

Food preferences vary significantly across cities, with different dishes dominating order volumes in different regions. This highlights strong regional taste patterns among Zomato users.

Business Recommendation

- ✓ Enables city-specific menu recommendations and personalized home screens
- ✓ Assists restaurant partners in menu optimization based on city demand

Customer Churn Analysis

❓ Business Question

Identify customers on the Zomato platform who were **active in 2024** but have not placed any **orders in 2025**, to analyze potential customer churn.

</>SQL Query

```
SELECT DISTINCT
c.customer_id,
c.customer_name
FROM orders o
JOIN customers c
ON c.customer_id = o.customer_id
WHERE EXTRACT(YEAR FROM o.order_date) = 2024
AND NOT EXISTS (
SELECT 1
FROM orders o2
WHERE o2.customer_id = o.customer_id
AND EXTRACT(YEAR FROM o2.order_date) = 2025
)
ORDER BY 1;
```

☰ Output Summary

	AZ ↗ customer_id	AZ ↗ customer_name
1	C1028	Simon Bansal
2	C1059	Sudiksha Biswas
3	C1084	Fariq Barad
4	C1090	Mugdha Pandit
5	C1228	Tanish Murty
6	C1262	Sai Madan
7	C1326	Dhruv Master
8	C1365	Eiravati Madan
9	C1366	Yash Garg
10	C1387	Ishwar Sheth

💡 Key Insight

A significant set of customers who were previously active did not return in the following year, indicating potential churn driven by experience, pricing sensitivity, or competitive alternatives.

💼 Business Recommendation

- ✓ These users can be targeted with win-back campaigns, personalized offers, or reactivation notifications.
- ✓ Helps Zomato quantify year-over-year retention gaps and prioritize retention strategies over acquisition.

Year-over-Year Cancellation Rates

Business Question

Analyze and compare **year-over-year order cancellation rates** for restaurants on the Zomato platform to identify operational performance trends

SQL Query

```

WITH cancellation_data_25 AS (
SELECT
o.restaurant_id,
r.restaurant_name,
COUNT(o.order_id) AS total_orders,
SUM(
CASE
WHEN d.delivery_id IS NULL THEN 1
ELSE 0
END) AS cancelled_orders
FROM orders o
LEFT JOIN deliveries d
ON o.order_id = d.order_id
LEFT JOIN restaurants r
ON o.restaurant_id = r.restaurant_id
WHERE YEAR(o.order_date) = 2025
GROUP BY
o.restaurant_id,
r.restaurant_name
),
Last_Year_Data AS (
SELECT
restaurant_id,
restaurant_name,
total_orders,
cancelled_orders,
ROUND((cancelled_orders / NULLIF(total_orders, 0)) * 100, 2)
AS cancellation_rate_2025
FROM cancellation_data_25,
cancellation_data_26 AS (
SELECT
o.restaurant_id,
r.restaurant_name,
COUNT(o.order_id) AS total_orders,
SUM(
CASE
WHEN d.delivery_id IS NULL THEN 1
ELSE 0
END) AS cancelled_orders
FROM orders o
LEFT JOIN deliveries d
ON o.order_id = d.order_id
LEFT JOIN restaurants r
ON o.restaurant_id = r.restaurant_id
WHERE YEAR(o.order_date) = 2026
GROUP BY
o.restaurant_id,
r.restaurant_name ),
Current_Year_Data AS (
SELECT
restaurant_id,
restaurant_name,
total_orders,
cancelled_orders,
ROUND((cancelled_orders / NULLIF(total_orders, 0)) * 100, 2)
AS cancellation_rate_2026
FROM cancellation_data_26)
SELECT
l.restaurant_id,
l.restaurant_name,
l.cancellation_rate_2025,
c.cancellation_rate_2026,
ROUND(
c.cancellation_rate_2026 - l.cancellation_rate_2025,
2) AS rate_difference
FROM Last_Year_Data l
LEFT JOIN Current_Year_Data c
ON l.restaurant_id = c.restaurant_id;

```

Output Summary

	A2 restaurant_id	A2 restaurant_name	I23 cancellation_rate_2025	I23 cancellation_rate_2026	I23 rate_difference
1	R78	Lajitas Dhaba	7.17	9.82	2.65
2	R99	Cafeibz Grill	8.96	10.56	1.6
3	R87	Lukes Resto	9.57	9.09	-0.48
4	R5	Nehas Bistro	8.04	11.65	3.61
5	R57	Imarans Point	7.63	10.43	2.8
6	R102	Dhrus Dhaba	9.09	8.48	-0.61
7	R47	Zaids Cafe	9.51	6.85	-2.66
8	R36	Harshs Kitchen	11.14	9.69	-1.71
9	R72	Poojas Grill	6.37	8.45	2.08
10	R85	Nirjas Kitchen	11.63	8.42	-3.21
11	R31	Benjamins Grill	7.73	8.78	1.05
12	R29	Yagneshs Grill	6.64	10.23	3.59
13	R92	Gayathris Grill	11.16	10.02	-1.14
14	R30	Chasmuns Dhaba	9.13	8.42	-0.71
15	R41	Henrys Kitchen	8.73	14.52	5.79
16	R43	Nehas Bistro	13.23	12.31	-0.92
17	R96	Hardiks Cafe	9.27	9.47	0.2
18	R67	Gabriels Cafe	8.06	7.51	-0.55
19	R51	Jagdishs Point	15.89	10.74	-5.15
20	R68	Eshas Cafe	10.4	11.13	0.73
21	R97	Yashs Resto	10.22	9.19	-1.03

Key Insight

Cancellation rates vary significantly across restaurants year-over-year, with some showing clear improvement while others experience an increase. This highlights inconsistent operational reliability at the restaurant level.

Business Recommendation

- These users can be targeted with win-back campaigns, personalized offers, or reactivation notifications.
- Cancellation rate trends directly impact customer trust, ratings, and retention.

Rider Delivery Performance

Business Question

Calculate the average delivery time for each delivery partner (rider) on the Zomato platform to evaluate rider performance and delivery efficiency..

SQL Query

```

SELECT
    r.rider_id,
    r.rider_name,
    SEC_TO_TIME(
        AVG(
            TIMESTAMPDIFF(
                SECOND,
                CONCAT(o.order_date, ' ', o.order_time),
                CONCAT(o.order_date, ' ', d.delivery_time)
            )
        )
    ) AS avg_delivery_time
FROM riders r
JOIN deliveries d
ON r.rider_id = d.rider_id
JOIN orders o
ON d.order_id = o.order_id
WHERE d.delivery_status = 'Delivered'
GROUP BY
    r.rider_id,
    r.rider_name
ORDER BY
    avg_delivery_time;

```

Output Summary

	AZ ↗ rider_id	AZ rider_name	⌚ avg_delivery_time
1	RD24	Charles Bhandari	00:33:37.0491
2	RD195	Charan Atwal	00:33:39.2783
3	RD179	Dalbir Deshpande	00:33:40.1652
4	RD16	Triya Kari	00:33:42.2950
5	RD97	Manan Bhattacharyya	00:33:50.0591
6	RD158	Onkar Bandi	00:33:51.1764
7	RD192	Waida Brahmbhatt	00:33:52.9906
8	RD208	Arunima Nagy	00:33:57.6106
9	RD131	Qasim Narayan	00:33:58.0830
10	RD144	Chandresh Narayan	00:33:58.0172

Key Insight

Average delivery times are largely consistent across riders, with a small performance gap between the fastest and slowest delivery partners. This indicates a generally stable last-mile delivery operation with minor efficiency variations.

Business Recommendation

- ✓ High-performing riders can be prioritized during peak demand periods and rewarded through incentive programs.
- ✓ Slightly slower riders can be supported with route optimization, workload balancing, or targeted training.

Restaurant Growth Trends

Business Question

Analyze the **month-over-month growth ratio** of delivered orders for each restaurant on the Zomato platform to evaluate demand trends and restaurant performance over time.

SQL Query

```

WITH monthly_orders AS ( SELECT
o.restaurant_id,
MAX(r.restaurant_name) AS restaurant_name,
DATE_FORMAT(o.order_date, '%m-%y') AS month,
COUNT(o.order_id) AS cnt_orders,
STR_TO_DATE(DATE_FORMAT(o.order_date, '%Y-%m-01'), '%Y-%m-%d')
AS sort_month
FROM orders o
JOIN deliveries d
ON d.order_id = o.order_id
JOIN restaurants r
ON r.restaurant_id = o.restaurant_id
WHERE d.delivery_status = 'Delivered'
GROUP BY
o.restaurant_id,
month,
sort_month),
final_data AS ( SELECT
restaurant_id,
restaurant_name,
month,
cnt_orders AS current_month_orders,
sort_month,

```

```

LAG(cnt_orders, 1) OVER (
PARTITION BY restaurant_id
ORDER BY sort_month
) AS prev_month_orders
FROM monthly_orders)
SELECT
restaurant_id,
restaurant_name,
month,
current_month_orders,
prev_month_orders,
ROUND(
(current_month_orders / NULLIF(prev_month_orders, 0) -
1),
2) AS growth_ratio
FROM final_data
ORDER BY restaurant_id, sort_month;

```

Output Summary

	AZ restaurant_id	AZ restaurant_name	AZ month	I23 current_month_orders	I23 prev_month_orders	I23 growth_ratio
1	R1	Vrindas Bistro	03-22	1	[NULL]	[NULL]
2	R1	Vrindas Bistro	05-22	2	1	1
3	R1	Vrindas Bistro	07-22	1	2	-0.5
4	R1	Vrindas Bistro	08-22	3	1	2
5	R1	Vrindas Bistro	09-22	4	3	0.33
6	R1	Vrindas Bistro	10-22	2	4	-0.5
7	R1	Vrindas Bistro	11-22	1	2	-0.5
8	R1	Vrindas Bistro	12-22	5	1	4
9	R1	Vrindas Bistro	01-23	2	5	-0.6
10	R1	Vrindas Bistro	02-23	6	2	2
11	R1	Vrindas Bistro	03-23	1	6	-0.83
12	R1	Vrindas Bistro	04-23	6	1	5
13	R1	Vrindas Bistro	05-23	3	6	-0.5
14	R1	Vrindas Bistro	06-23	3	3	0
15	R1	Vrindas Bistro	07-23	5	3	0.67
16	R1	Vrindas Bistro	08-23	4	5	-0.2
17	R1	Vrindas Bistro	09-23	6	4	0.5
18	R1	Vrindas Bistro	10-23	5	6	-0.17

Refresh Save Cancel Export data 100000 5,781 5781 row(s) fetched - 1.908s (0.023s fe)

Key Insight

Monthly order volumes for restaurants show fluctuating growth patterns, with periods of both acceleration and decline. This indicates variability in demand influenced by seasonality, promotions, and restaurant-level operational performance.

Business Recommendation

- ✓ Restaurants with sustained positive growth can be prioritized for premium placements and marketing campaigns.
- ✓ Monitoring growth ratios enables early detection of declining partners and supports proactive retention and revenue optimization strategies.

Customer Segmentation

Business Question

Segment customers on the Zomato platform into '**Gold**' and '**Silver**' categories based on their total spending compared to the overall Average Order Value (AOV), and analyze each segment's order volume and revenue contribution.

SQL Query

```
WITH customer_spend AS (
  SELECT
    c.customer_id,
    c.customer_name,
    SUM(o.total_amount) AS total_spent,
    COUNT(o.order_id) AS total_orders
  FROM customers c
  JOIN orders o
  ON c.customer_id = o.customer_id
  WHERE o.order_status = 'Delivered'
  GROUP BY
    c.customer_id,
    c.customer_name
),
avg_order_value AS (
  SELECT
    AVG(total_amount) AS aov
  FROM orders
  WHERE order_status = 'Delivered'
)
```

```
SELECT
  cs.customer_id,
  cs.customer_name,
  cs.total_orders,
  cs.total_spent,
  CASE
    WHEN cs.total_spent >= aov THEN 'Gold'
    ELSE 'Silver'
  END AS segment
FROM customer_spend cs
CROSS JOIN avg_order_value
ORDER BY cs.total_spent DESC;
```

Output Summary

	AZ customer_id	AZ customer_name	123 total_orders	123 total_spent	AZ segment
1	C5754	Ekiya Sarin	21	5,331.06	Gold
2	C2197	Jagat Khosla	21	4,984.66	Gold
3	C2869	Ati Gokhale	21	4,839.54	Gold
4	C743	Bishakha Lal	20	4,780.04	Gold
5	C8162	Dev Banerjee	21	4,692.79	Gold
6	C1909	Ayush Reddy	20	4,679.33	Gold
7	C767	Varenaya Bhat	19	4,603.14	Gold
8	C9779	Chameli Shenoy	18	4,486.38	Gold
9	C635	Pushti Ramachandran	17	4,484.7	Gold
10	C5777	Yashawini Sane	19	4,470.44	Gold

Key Insight

The '**Gold**' customer segment consists of users with consistently high order frequency and total spending, indicating strong engagement and repeat purchasing behavior. Although this segment represents a smaller portion of the overall customer base, it contributes a disproportionately high share of total revenue.

Business Recommendation

- ✓ '**Gold**' customers should be prioritized for premium loyalty programs, personalized offers, and priority delivery benefits to maximize retention and lifetime value.
- ✓ Behavioral patterns from Gold customers can be used to design upsell strategies and convert high-potential Silver customers into the Gold segment.

Rider Monthly Earnings

Business Question

Calculate the monthly earnings of each delivery partner (rider) on the Zomato platform, assuming riders earn a fixed 8% commission per delivered order, to analyze income distribution and rider productivity over time.

SQL Query

```

SELECT
    r.rider_id,
    r.rider_name,
    DATE_FORMAT(o.order_date, '%Y-%m') AS month_year,
    SUM(o.total_amount * 0.08) AS monthly_earnings
FROM riders r
JOIN deliveries d
ON r.rider_id = d.rider_id
JOIN orders o
ON d.order_id = o.order_id
WHERE d.delivery_status = 'Delivered'
GROUP BY
    r.rider_id,
    r.rider_name,
    month_year
ORDER BY
    3,4;
  
```

Output Summary

	AZ rider_id	AZ rider_name	AZ month_year	I23 monthly_earnings
Grid	1 RD213	Veda Mand	2022-01	19.628
Text	2 RD44	Ranveer Cheema	2022-01	22.9592
Record	3 RD250	Damini Bahri	2022-01	43.0688
Record	4 RD165	Chatura Varty	2022-01	51.312
Record	5 RD145	Forum Kara	2022-01	158.2832
Record	6 RD165	Chatura Varty	2022-02	12.7104
Record	7 RD161	Yuvraj Srinivasan	2022-02	27.8104
Record	8 RD174	Ria Sura	2022-02	28.0496

Refresh Save Cancel Export data 1000000 8,200

Key Insight

Monthly earnings of riders vary significantly, reflecting differences in order volume, delivery frequency, and rider engagement levels. Some riders consistently generate higher earnings, indicating stronger productivity and availability.

Business Recommendation

- ✓ High-earning riders can be identified as top performers and rewarded through incentive programs or priority order allocation.
- ✓ Riders with lower or inconsistent monthly earnings may benefit from improved shift allocation, zone optimization, or performance support.

Rider Rating Distribution

Business Question

Analyze rider performance on the Zomato platform by categorizing deliveries into 5-star, 4-star, and 3-star ratings based on delivery time thresholds, and calculate the distribution of ratings for each rider.

SQL Query

```
WITH delivery_time AS (
SELECT
r.rider_id,
r.rider_name,
SEC_TO_TIME(
TIME_TO_SEC(d.delivery_time) -
TIME_TO_SEC(o.order_time)
) AS total_time
FROM orders o
JOIN deliveries d
ON o.order_id = d.order_id
JOIN riders r
ON d.rider_id = r.rider_id
),
rider_rating as
(SELECT
rider_id,
rider_name,
total_time,
```

```
CASE
WHEN (TIME_TO_SEC(total_time) / 60) <
25 THEN '5 star'
WHEN (TIME_TO_SEC(total_time) / 60) >=
25
AND (TIME_TO_SEC(total_time) / 60) < 35
THEN '4 star'
ELSE '3 star'
END AS riders_rating
FROM delivery_time
select rider_id,
rider_name,
riders_rating,
COUNT(riders_rating) rating_count
from rider_rating
group by 1,2,3
order by 2;
```

Output Summary

	AZ rider_id	AZ rider_name	AZ riders_rating	123 rating_count
1	RD219	Aachal Magar	5 star	28
2	RD219	Aachal Magar	4 star	86
3	RD219	Aachal Magar	3 star	131
4	RD223	Aachal Parmer	3 star	200
5	RD223	Aachal Parmer	5 star	63
6	RD223	Aachal Parmer	4 star	126
7	RD191	Aadhya Bhandari	3 star	205
8	RD191	Aadhya Bhandari	5 star	58

Refresh Save Cancel | < < > > | Export data 1000000 749

Key Insight

Rider ratings based on delivery time show noticeable variation across delivery partners, with some riders consistently achieving higher ratings while others receive a larger share of lower ratings. This reflects differences in delivery efficiency, route familiarity, and workload management.

Business Recommendation

- ✓ Riders with a higher proportion of 5-star ratings can be identified as top performers and rewarded through incentives or priority order allocation..
- ✓ Riders with frequent 3-star ratings can be supported with route optimization, zone reassignment, or targeted training programs.

Restaurant Peak Days

Business Question

Analyze day-wise order frequency for each restaurant on the Zomato platform and identify the peak ordering day of the week to understand demand patterns.

SQL Query

```

WITH restaurent_freq_day AS (
SELECT
r.restaurant_id,
r.restaurant_name,
DAYNAME(o.order_date) AS days,
COUNT(o.order_id) AS total_order,
RANK() OVER (PARTITION BY r.restaurant_id ORDER BY COUNT(o.order_id)
DESC) AS day_rank
FROM orders o
JOIN restaurants r
ON o.restaurant_id = r.restaurant_id
GROUP BY
1, 2, 3)
SELECT
restaurant_id,
restaurant_name,
days,
total_order
FROM restaurent_freq_day
WHERE day_rank = 1
ORDER BY
1, 4 DESC;
    
```

Output Summary

	AZ restaurant_id	AZ restaurant_name	AZ days	123 total_order
Grid	R1	Vrindas Bistro	Wednesday	157
Text	R10	Gangas Grill	Monday	150
Record	R100	Ojasvis House	Wednesday	158
Record	R101	Jeremias Kitchen	Wednesday	142
Record	R102	Dhrufs Dhaba	Monday	158
Record	R103	Darpans Bistro	Wednesday	157
Record	R11	Victors Resto	Thursday	154
Record	R12	Nidhis House	Tuesday	153

Refresh Save Cancel Add Edit Delete First Previous Next Last Export data 1000000 108

Key Insight

Order frequency varies significantly across days of the week for different restaurants, with each restaurant exhibiting a distinct peak ordering day. This indicates that customer demand is not uniform and is influenced by weekday-specific consumption patterns.

Business Recommendation

- ✓ Restaurants can optimize staffing, inventory, and preparation capacity based on their identified peak days.
- ✓ Zomato can schedule targeted promotions and push notifications on high-demand days to maximize order volume.

Business Question

Calculate the Customer Lifetime Value (CLV) for each customer on the Zomato platform by measuring the total revenue generated across all their orders, and identify high-value customers.

</>SQL Query

```
SELECT
c.customer_id,
c.customer_name,
SUM(o.total_amount) AS total_revenue
FROM customers c
JOIN orders o
ON c.customer_id = o.customer_id
GROUP BY
c.customer_id,
c.customer_name
ORDER BY
total_revenue DESC;
```

Output Summary

Grid

Text

Record

File

	AZ customer_id	AZ customer_name	123 total_revenue
1	C5754	Ekiya Sarin	5,571.29
2	C8162	Dev Banerjee	5,173.26
3	C2197	Jagat Khosla	5,162.91
4	C8116	Samaksh Krishnan	5,130.71
5	C2869	Ati Gokhale	4,983.11
6	C3710	Theodore Sidhu	4,946.95
7	C8073	Chandani Nagi	4,931.99
8	C4505	Yamini Muni	4,885.65

Key Insight

Customer lifetime value varies significantly across users, with a small group of customers contributing a disproportionately large share of total revenue. These high-CLV customers demonstrate strong loyalty and repeat purchasing behavior.

Business Recommendation

- ✓ High-CLV customers should be prioritized for retention through loyalty programs, personalized offers, and premium benefits.
 - ✓ Protecting this segment from churn is critical, as losing even a small number of high-CLV users can materially impact overall revenue.

Monthly Sales Trends

Business Question

Analyze month-over-month sales performance on the Zomato platform by comparing each month's total revenue with the previous month to identify growth or decline trends.

SQL Query

```

SELECT
    month_year,
    prev_month_rev,
    current_month_revenue,
    CASE
        WHEN prev_month_rev IS NULL THEN 'No Previous Data'
        WHEN current_month_revenue > prev_month_rev THEN 'Increase'
        WHEN current_month_revenue < prev_month_rev THEN 'Decrease'
        ELSE 'No Change'
    END AS sales_trend
FROM (
    SELECT
        DATE_FORMAT(o.order_date, '%Y-%m') AS month_year,
        SUM(o.total_amount) AS current_month_revenue,
        LAG(SUM(o.total_amount), 1) OVER (
            ORDER BY DATE_FORMAT(o.order_date, '%Y-%m')
        ) AS prev_month_rev
    FROM orders o
    GROUP BY month_year
) t
ORDER BY month_year;

```

Output Summary

	month_year	prev_month_rev	current_month_revenue	sales_trend
1	2022-01	[NULL]	3,899.52	No Previous Data
2	2022-02	3,899.52	8,359.69	Increase
3	2022-03	8,359.69	12,897.77	Increase
4	2022-04	12,897.77	19,326.37	Increase
5	2022-05	19,326.37	30,334.01	Increase
6	2022-06	30,334.01	33,708.78	Increase
7	2022-07	33,708.78	37,028.5	Increase
8	2022-08	37,028.5	40,604.95	Increase

Refresh Save Cancel Export data 1000000 60

Key Insight

Monthly sales exhibit a consistent upward trend, with revenue increasing steadily across consecutive months. The lack of significant declines indicates strong customer demand and sustained platform engagement.

Business Recommendation

- ✓ Consistent revenue growth reflects effective customer retention, successful restaurant partnerships, and stable delivery operations.
- ✓ This positive momentum allows Zomato to confidently scale marketing efforts, onboard new restaurants, and expand delivery capacity.

Rider Efficiency Analysis

Business Question

Evaluate rider efficiency on the Zomato platform by analyzing average delivery times and identifying the most efficient (fastest) and least efficient (slowest) delivery partners.

SQL Query

```
WITH avg_time AS (
SELECT
r.rider_id,
r.rider_name,
AVG(
TIME_TO_SEC(d.delivery_time) -
TIME_TO_SEC(o.order_time)
) AS avg_delivery_seconds
FROM orders o
JOIN deliveries d
ON o.order_id = d.order_id
JOIN riders r
ON r.rider_id = d.rider_id
GROUP BY
r.rider_id,
r.rider_name
)
```

```
SELECT
rider_id,
rider_name,
SEC_TO_TIME(avg_delivery_seconds) AS
avg_delivery_time,
CASE
WHEN avg_delivery_seconds = (
SELECT MIN(avg_delivery_seconds) FROM
avg_time
) THEN 'Most Efficient (Fastest)'
WHEN avg_delivery_seconds = (
SELECT MAX(avg_delivery_seconds) FROM
avg_time
) THEN 'Least Efficient (Slowest)'
ELSE 'Average'
END AS rider_efficiency
FROM avg_time;
```

Output Summary

	AZ \rightarrow rider_id	AZ rider_name	avg_delivery_time	AZ rider_efficiency
1	RD1	Bachittar Brahmbhatt	00:35:18.7387	Average
2	RD10	Andrew Hayer	00:34:56.1217	Average
3	RD100	Divya Andra	00:34:59.4958	Average
4	RD101	Ekiya Mani	00:34:28.6640	Average
5	RD102	Tanay Soni	00:35:28.6391	Average
6	RD103	Bina Sandal	00:34:43.1068	Average
7	RD104	Advik Taneja	00:34:15.0000	Average
8	RD105	Veda Wason	00:34:23.4568	Average

Key Insight

Average delivery times across riders fall within a relatively narrow range, indicating consistent delivery performance overall. However, distinct fastest and slowest riders can still be identified, highlighting measurable efficiency differences at the individual level.

Business Recommendation

- ✓ Most efficient riders can be rewarded with incentives and prioritized during peak demand periods.
- ✓ Least efficient riders can be supported through route optimization, zone reassignment, or performance coaching..

Seasonal Food Popularity

Business Question

Analyze the seasonal popularity of food items on the Zomato platform by identifying the top-ordered items in each season and measuring their order volume and revenue contribution.

SQL Query

```

WITH season_cte AS (
  SELECT
    o.order_id, o.order_item,
    o.total_amount,
    CASE
      WHEN MONTH(o.order_date) BETWEEN
        4 AND 7 THEN 'spring'
      WHEN MONTH(o.order_date) > 8
      AND MONTH(o.order_date) <= 10 THEN
        'summer'
      ELSE 'winter'
    END AS season
  FROM orders o
), season_summary AS (
  SELECT
    season, order_item,
    COUNT(order_id) AS total_orders,
    ROUND(SUM(total_amount), 0) AS
    total_revenue
  FROM season_cte
  GROUP BY
    season,
    order_item),
ranks AS (
  SELECT
    *,
    RANK() OVER (
      PARTITION BY season
      ORDER BY total_orders DESC
    ) AS rank_by_season
  FROM season_summary
)
SELECT
  season,
  order_item,
  total_orders,
  total_revenue
FROM ranks
WHERE rank_by_season BETWEEN 1 AND
  3;
  
```

Output Summary

	AZ season	AZ order_item	123 total_orders	123 total_revenue
1	spring	Butter Chicken	2,534	876,030
2	spring	Dal Makhani	2,520	590,173
3	spring	Burger & Fries	2,503	464,052
4	summer	Idli Sambar	1,661	163,824
5	summer	Tandoori Roti	1,650	82,502
6	summer	Veg Biryani	1,638	364,477
7	winter	Tandoori Roti	4,348	217,404
8	winter	Pizza Margherita	4,345	1,338,279

Refresh : Save : Cancel : |< < > >| : Export data : 1000000 : 9

Key Insight

Food item popularity shows clear seasonal variation, with different dishes dominating order volumes in spring, summer, and winter. This indicates that customer food preferences are strongly influenced by seasonal factors.

Business Recommendation

- ✓ Restaurants can optimize menus and inventory by promoting seasonally high-demand items.
- ✓ Zomato can design season-specific campaigns and personalized recommendations to increase order conversion.

Customer Ordering Consistency

Business Question

Evaluate customer ordering consistency on the Zomato platform by measuring active ordering months, average order frequency, and average gap between orders to classify customers into reliability segments.

SQL Query

```
WITH base_orders AS (
  SELECT
    c.customer_id, c.customer_name,
    o.order_date,
    DATE_FORMAT(o.order_date, '%Y-%m') AS order_month,
    LAG(o.order_date) OVER (
      PARTITION BY c.customer_id
      ORDER BY o.order_date
    ) AS prev_order_date
  FROM orders o
  JOIN customers c
  ON c.customer_id = o.customer_id),
  gap_data AS (SELECT
    customer_id, customer_name,
    order_date, order_month,
    CASE
      WHEN prev_order_date IS NULL THEN NULL
      ELSE DATEDIFF(order_date, prev_order_date)
    END AS gap_days
  FROM base_orders
  ),
  customer_metrics AS (SELECT
    customer_id, customer_name,
    COUNT(DISTINCT order_month) AS active_months,
    COUNT(order_date) AS total_orders, ROUND(
      COUNT(order_date) / COUNT(DISTINCT order_month), 2) AS avg_orders_per_month,
    ROUND(AVG(gap_days), 2) AS avg_gap_days
  FROM gap_data
  GROUP BY
    customer_id,
    customer_name
  )
  SELECT
    customer_id, customer_name,
    active_months, avg_orders_per_month,
    avg_gap_days,
    CASE
      WHEN active_months >= 6
      AND avg_gap_days <= 15 THEN 'High Reliability'
      WHEN active_months BETWEEN 3 AND 5
      AND avg_gap_days BETWEEN 16 AND 30 THEN 'Medium Reliability'
      ELSE 'Low Reliability'
    END AS reliability_score
  FROM customer_metrics
  ORDER BY customer_id;
```

Output Summary

	AZ customer_id	AZ customer_name	123 active_months	123 avg_orders_per_month	123 avg_gap_days	AZ reliability_score
Grid	1 C1	Anvi Konda	9	1.44	109.17	Low Reliability
Text	2 C10	Tripti Yadav	5	1.2	179.8	Low Reliability
Text	3 C100	Zinal Yohannan	7	1.43	40.89	Low Reliability
Text	4 C1000	Shrawya Ravel	9	1.22	50.5	Low Reliability
Text	5 C10000	Saksham Sahni	9	1.11	110.22	Low Reliability
Text	6 C1001	Anay Ramachandran	5	1	95.75	Low Reliability
Text	7 C1002	Dev Bath	7	1.14	91.71	Low Reliability
Text	8 C1003	Yadavi Cherian	8	1	194	Low Reliability

Key Insight

A large proportion of customers fall into the 'Low Reliability' segment, characterized by long gaps between orders and relatively low monthly ordering frequency. This indicates sporadic engagement rather than habitual usage of the platform.

Business Recommendation

- ✓ Low-reliability customers represent a significant reactivation opportunity through personalized reminders, discounts, and reorder nudges.
- ✓ Medium and high-reliability customers can be targeted with loyalty programs and subscription-based benefits to reinforce consistent ordering behavior.



Key Insights Summary

1

Customer Loyalty Drives Revenue

High-frequency customers (>20 orders) demonstrate consistently higher AOV, contributing disproportionately to total revenue. Retention strategies yield greater ROI than acquisition.

2

Peak Hours Present Optimization Opportunities

Order volume peaks during 14:00-16:00 and 20:00-22:00 time slots. Dynamic pricing and optimized delivery partner allocation during these windows can significantly improve margins.

3

VIP Customers Require Special Attention

A small group of high-value customers (spending >₹5,000) drive disproportionate revenue. Losing even a few VIP customers materially impacts overall platform performance.

4

Operational Issues Are Restaurant-Specific

Non-delivered orders and cancellation rates cluster around specific restaurants, indicating systemic operational issues that require targeted audits and partner support.

5

Regional Preferences Drive Local Success

Each city has distinct top-revenue restaurants and popular dishes, highlighting the importance of localized strategies rather than one-size-fits-all approaches.

6

Rider Performance Is Measurable and Actionable

Delivery times and ratings vary across riders, enabling data-driven incentive programs, targeted training, and optimized order allocation for improved customer satisfaction.

7

Churn Prevention Is Critical

Significant customer segments show low reliability with long gaps between orders. Proactive reactivation campaigns and personalized offers can convert occasional users into loyal customers.

Overall Impact

These insights provide a comprehensive view of Zomato's business ecosystem, enabling data-driven decisions across customer retention, operational efficiency, revenue optimization, and partnership management.



Conclusion

Business Impact

This SQL analytics project delivers actionable insights across multiple business dimensions:

- ✓ **Customer Retention:** Identified high-value segments and churn patterns
- ✓ **Revenue Optimization:** Analyzed AOV, CLV, and spending patterns
- ✓ **Operational Excellence:** Evaluated rider performance and delivery efficiency
- ✓ **Partnership Strategy:** Assessed restaurant performance and growth trends

</> Skills Demonstrated

Advanced SQL

CTEs, Window Functions, JOINs

Data Analysis

Trend Analysis, Segmentation

Business Intelligence

KPIs, Metrics, Insights

Problem Solving

Real-world Business Questions

★ Why This Project Is Interview-Ready

- ✓ **Real Dataset:** Analysis based on actual business data from Zomato
- ✓ **Complex Queries:** Demonstrates mastery of advanced SQL techniques
- ✓ **Business Context:** Every query tied to actionable business recommendations
- ✓ **End-to-End:** From data extraction to insights to strategic recommendations
- ✓ **FAANG-Ready:** Follows industry best practices for data analytics

👉 Next Steps

This project can be extended with:

- Predictive modeling for churn prediction
- Dashboard visualization in Tableau/Power BI
- A/B testing framework for recommendations

Thank You

Questions? Let's discuss the insights and SQL techniques!