ONLINE FOOD DELIVERY BUSINESS INTELLIGENCE & PERFORMANCE ANALYSIS

Inspired by real platforms like (Swiggy,Zomato, Uber Eats)

FEBRUARY 2026

Prepared by

Liza Deka

Table of Contents

INTRODUCTION

The Online Food Delivery Business Intelligence & Performance Analysis project focuses on analyzing transactional and operational data from a simulated food delivery platform. Using structured datasets covering customers, restaurants, orders, order items, and delivery agents, the project evaluates key performance indicators such as revenue generation, customer behavior, restaurant performance, and delivery efficiency.

This analysis is important because online food delivery businesses operate as multi-sided marketplaces where profitability depends on optimizing customer retention, restaurant partnerships, discount strategies, and operational speed. By examining order patterns, revenue trends, demographic segments, and delivery performance, the project identifies actionable insights that support data-driven decision-making.

SQL was used extensively to perform data extraction, aggregation, joins, KPI calculations, and trend analysis across relational tables. Through structured queries and performance metrics, the project demonstrates how raw transactional data can be transformed into meaningful business intelligence insights, reflecting real-world analytical scenarios in modern digital commerce platforms.

PROBLEM STATEMENT

Online food delivery platforms generate large volumes of transactional and operational data, but without structured analysis, this data cannot effectively support business decisions. Key challenges include understanding revenue drivers, identifying high-performing restaurants and menu items, evaluating customer ordering behavior, and monitoring delivery efficiency across cities.

This project aims to answer critical business questions such as: Which customers and restaurants contribute most to revenue? Which items and cuisines drive demand? How do discounts and delivery times impact order value and customer experience? Where do operational inefficiencies exist within the delivery network?

Solving these problems is essential for improving profitability, enhancing customer retention, optimizing operational performance, and enabling data-driven strategy decisions. By addressing these challenges through SQL-based analysis, the project demonstrates how business intelligence can be derived from relational data to support real-world food delivery platforms.

BUSINESS OBJECTIVES

As a Data Analyst for the Online Food Delivery platform, the project aims to achieve the following measurable business objectives:

- **Increase Revenue Performance** by identifying top-performing restaurants, high-revenue items, and optimizing Average Order Value (AOV).
- **Improve Customer Retention** by analyzing repeat purchase behavior, customer segmentation (city, gender), and identifying high-value customers.
- **Optimize Discount Strategy** by evaluating the impact of discounts on order volume and overall revenue contribution.
- **Enhance Restaurant Growth** by bench marking restaurant performance based on revenue, order volume, ratings, and cuisine popularity.
- **Strengthen Product Performance** by identifying top-selling items based on quantity and revenue contribution.
- **Improve Operational Efficiency** by analyzing average delivery time, agent performance, and city-level logistics trends.
- Support Data-Driven Decision Making through KPI tracking, trend analysis, and performance dashboards derived from SQL queries.
- Identify Growth Opportunities by evaluating time-based sales trends and uncovering under performing segments.

These objectives align analytical insights with strategic business outcomes, ensuring measurable impact across revenue, customer engagement, operational performance, and marketplace growth.

DATASET OVERVIEW

- Dataset Description

The dataset represents a simulated relational database of an Online Food Delivery platform. It captures transactional, customer, restaurant, and operational data required to perform business intelligence and performance analysis using SQL. The structured schema enables multi-dimensional analysis across revenue, customer behavior, product sales, and delivery efficiency.

- Tables Involved

The database consists of five primary tables:

- customers – Customer demographic and signup information
- restaurants – Restaurant partner details and ratings
- orders – Order-level transactional data
- order_item – Item-wise breakdown of each order
- delivery_agents – Delivery workforce information

- Record Size

The dataset contains structured records across multiple entities, enabling large-scale transactional analysis. It supports multi-table joins for revenue, performance, and operational insights.

- Time Period Covered

The data spans across multiple order dates from customer signup to delivery completion, enabling trend and time-based performance analysis

- Type of Data Stored

Structured, relational data including customer details, order transactions, menu pricing, and restaurant information.

ER DIAGRAM EXPLANATION

Entities

The database consists of five primary entities:

- Customers (customer_id as Primary Key)
- Restaurants (restaurant_id as Primary Key)
- Orders (order_id as Primary Key)
- Order_Item (order_item_id as Primary Key)
- Delivery_Agents (agent_id as Primary Key)

Relationships

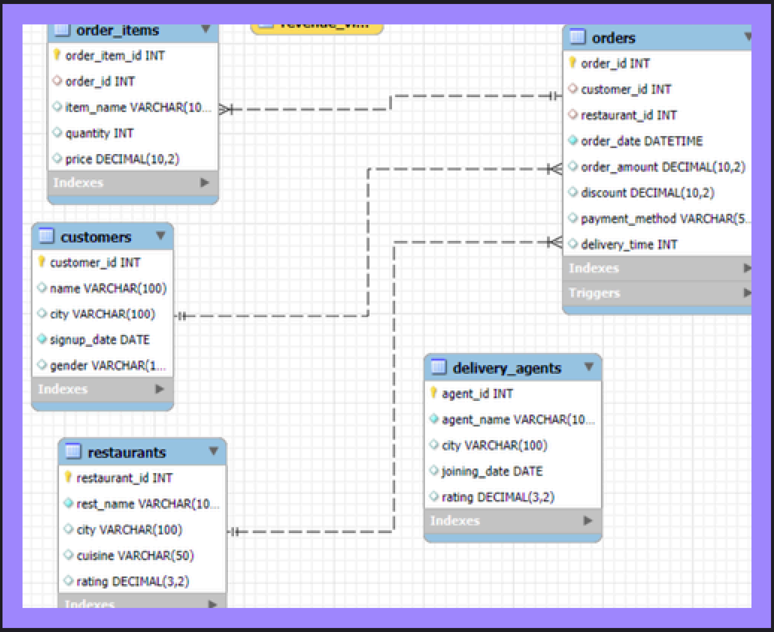
- A Customer can place multiple Orders.
- Each Order can contain multiple Menu_Items, linked through Order_Details.
- A Restaurant offers multiple Menu_Items.

Cardinality

- Customer → Orders: One-to-Many
- Restaurant → Menu_Items: One-to-Many
- Orders ↔ Menu_Items: Many-to-Many (resolved using Order_Details)

Key Constraints

- Each entity has a primary key (e.g., Customer_ID, Order_ID).
- Foreign keys ensure referential integrity between related tables.



EXPLORATORY DATA ANALYSIS (EDA)

1.**Total Revenue**

```
select sum(order_amount - discount) as total_revenue from orders;
```

1. Business Question

What is the total net revenue generated by the business after applying discounts across all customer orders?

2. Tables Used

- Orders – contains order-level transaction data, including order amount and discount applied.

Key columns used:

- order_amount → The gross value of the order before any discount
- discount → The discount applied to that order

Each row represents a single transaction.

The expression order_amount - discount calculates the net revenue for each individual order.

3. Business Insight Derived

The query calculates the net revenue by subtracting discounts from the order amount for each order and summing it across all orders. This provides a true picture of revenue realized, helping the business evaluate financial performance, pricing effectiveness, and the impact of discount strategies.

“This query calculates the total net revenue by summing order values after discounts, providing a clear view of the company’s actual earnings”

EXPLORATORY DATA ANALYSIS (EDA)

2. **Total Orders Per City**

```sql
select r.city, count(o.order_id) as total_orders
from orders o join restaurants r on o.restaurant_id =
r.restaurant_id
group by r.city
order by total_orders desc;
```

1. Business Question

What is the total number of orders placed in each city, and which cities generate the highest order volume?

2. Tables Used

- orders (o)

Key column used:

- order_id → Unique identifier for each order
- restaurant_id → Links each order to the restaurant that fulfilled it
- restaurants (r)

Key column used:

- restaurant_id → Primary key used for joining
- city → Location of the restaurant

3. Business Insight Derived

The query calculates the total number of orders per city and ranks cities in descending order of demand.

This insight can be used to:

- Allocate delivery partners and logistics resources efficiently
- Decide where to scale restaurant on boarding
- Optimize regional marketing and promotional campaigns
- Support city-wise revenue and growth forecasting

EXPLORATORY DATA ANALYSIS (EDA)

3. **Top 10 Customers by Spending**

```
select c.name, sum(o.order_amount - o.discount) as total_spent
from customers c join orders o on o.customer_id =
c.customer_id
group by c.customer_id order by total_spent desc
limit 10;
```

1. Business Question
Who are the top 10 highest-value customers based on their total net spending on the platform?

2. Tables Used

- Customers Table (customers)

customer_id → Unique identifier for each customer
name → Customer name used for business reporting

- Orders Table (orders)

customer_id → Links each order to a customer
order_amount → Gross value of the order
discount → Discount applied to the order

3. Business Insight Derived
The output identifies the top 10 highest-value customers based on net revenue contribution.
This insight enables the business to:

- Identify VIP customers who drive a significant share of revenue
- Improve retention through targeted loyalty programs and exclusive offers
- Optimize marketing by building look-alike campaigns based on high-value customer profiles
- Assess revenue concentration risk if earnings depend on a small customer group

CUSTOMER SEGMENTATION

Customer Category (Gold/Silver/Bronze)

```
select c.name, sum(o.order_amount - o.discount) as
total_spent,
case when sum(o.order_amount - o.discount) >= 5000 then
'Gold'
     when sum(o.order_amount - o.discount) >= 2000 then 'Silver'
     else 'Bronze' end as customer_category
from customers c
join orders o on o.customer_id = c.customer_id
group by c.customer_id;
```

1. Business Question
Which customers generate the highest net revenue for the business, and how can they be classified into Gold, Silver, and Bronze segments based on their total spending after discounts?

2. Tables Used

- customers

Contains customer details such as customer_id and name.

- orders

Stores transactional data including order_amount and discount.

3. Business Insight Derived
The result shows each customer’s total net spending and assigns them to a loyalty category:

- Gold: High-value customers (₹5000+)
- Silver: Medium-value customers (₹2000–₹4999)
- Bronze: Low-value customers (<₹2000)

This insight helps the business:

- Identify high-value customers
- Design targeted loyalty programs
- Prioritize retention and personalized marketing efforts

RESTAURANT PERFORMANCE ANALYSIS

1. **Top 10 Restaurants by Revenue**

```
select r.rest_name, r.city,
sum(o.order_amount - o.discount) as total_revenue
from restaurants r join orders o on o.restaurant_id =
r.restaurant_id
group by r.restaurant_id, r.city
order by total_revenue desc
limit 10
;
```

1. Business Question

Which are the top 10 highest-revenue restaurants on the platform based on net revenue (order amount minus discount)?

2. Tables Used

- restaurants → Provides restaurant details such as name and city
- orders → Contains transactional data including order amount and discounts

The tables are joined using restaurant_id, and net revenue is calculated per restaurant.

3. Business Insight Derived

The query returns the top 10 restaurants by net revenue, ranked from highest to lowest.

This insight helps the business:

- Identify top-performing restaurant partners
- Support partner retention and incentive programs
- Analyze city-wise revenue concentration
- Make data-driven decisions for promotions and partnerships

RESTAURANT PERFORMANCE ANALYSIS

2. **Average Rating vs Revenue**

```
select r.rest_name, round(avg(r.rating),2) as avg_rating,
sum(o.order_amount - o.discount) as total_revenue
from restaurants r join orders o on o.restaurant_id =
r.restaurant_id
group by r.restaurant_id, r.city
order by total_revenue desc;
```

1. Business Question
What is the average customer rating and total net revenue generated by each restaurant, and how do restaurants rank when ordered by revenue?

2. Tables Used

- restaurants

Contains restaurant details such as:
restaurant_id, rest_name, rating, city

- orders

Contains order-level transaction data:
order_amount, discount, restaurant_id
The tables are joined using restaurant_id to combine operational and transactional data.

3. Business Insight Derived

- Identifies top revenue-generating restaurants
- Shows whether higher ratings translate into higher revenue
- Helps management:
    - Reward high-performing restaurants
    - Detect cases where low-rated restaurants still generate high revenue
    - Improve partner strategy and quality control

This analysis supports restaurant performance evaluation and partner optimization.

DELIVERY PERFORMANCE ANALYSIS

1. **Average Delivery Time Per City**

```
select r.city,
round(avg(o.delivery_time),2) as avg_delivery_time
from restaurants r join orders o on o.restaurant_id =
r.restaurant_id
group by r.city
order by avg_delivery_time;
```

1. Business Question
What is the average delivery time in each city, and how do cities compare in terms of delivery efficiency?

2. Tables Used

- restaurants: Provides city-level information for each restaurant
- orders: Contains delivery time data for individual orders

The tables are joined using restaurant_id to link each order to its city.

3. Business Insight Derived
The query calculates the average delivery time per city, rounded to two decimal places, and ranks cities from fastest to slowest delivery.
This insight helps the business:

- Identify cities with delivery bottlenecks
- Benchmark operational performance across locations
- Take data-driven actions to improve logistics and customer experience

DELIVERY PERFORMANCE ANALYSIS

2. **Late Deliveries (Above 45 Minutes)**

```
select r.city, round(avg(o.delivery_time),2) as avg_delivery_time
from restaurants r join orders o on o.restaurant_id =
r.restaurant_id group by r.city having
round(avg(o.delivery_time),2) > 45 ;
```

1. Business Question
Which cities are experiencing late deliveries, where the average delivery time exceeds 45 minutes?

2. Tables Used

- restaurants

Contains restaurant details, including city and restaurant_id.

- orders

Stores order-level information such as delivery_time and restaurant_id.
The tables are joined using restaurant_id to associate each order with its city.

.

3. Business Insight Derived
The query highlights cities with poor delivery performance by filtering those whose average delivery time is greater than 45 minutes.
This insight helps the business:

- Identify operational bottlenecks by city
- Improve logistics and delivery partner allocation
- Set targeted actions to reduce delays and improve customer satisfaction

VIEWS

Create Revenue View

```
create view revenue_view as
select
r.restaurant_id,
r.rest_name,
r.city,
sum(o.order_amount - o.discount) as total_revenue
from restaurants r join orders o on r.restaurant_id =
o.restaurant_id
group by r.restaurant_id, r.rest_namer.city
order by total_revenue desc;
select * from revenue_view;
```

1. Business Question

Which restaurants are generating the highest net revenue (after discounts), and how do their revenues compare across cities?

2. Tables Used

- restaurants

Contains restaurant-level information such as restaurant ID, name, and city.

- orders

Contains transactional data including order amount and discount.

The query joins these tables using restaurant_id.

3. Business Insight Derived

The created view provides a ranked list of restaurants by total net revenue, making it easy to:

- Identify top-performing restaurants
- Compare revenue performance across cities
- Support reporting, dashboards, and recurring revenue analysis

This view simplifies repeated analysis and serves as a reusable revenue KPI layer for business decision-making.

STORED PROCEDURES

Get Top N Restaurant

```
CREATE PROCEDURE GetTopNRestaurants(IN top_n INT)
BEGIN
    SELECT r.rest_name, r.rating, count(o.order_id) as
total_orders
    FROM restaurants r
    join orders o on o.restaurant_id = r.restaurant_id
    group by r.restaurant_id
    ORDER BY total_orders  DESC
    LIMIT top_n;
END;
call GetTopNRestaurants(10);
```

1. Business Question

Which restaurants are the top-performing on the platform based on order volume, and how do they rank when we consider customer engagement and ratings?

2. Tables Used

- restaurants

Contains restaurant-level information such as restaurant name and customer rating.

- orders

Stores order transaction data and is used to count how many orders each restaurant has received.

These tables are joined using the restaurant_id.

3. Business Insight Derived

The procedure identifies the Top N restaurants with the highest number of orders, indicating strong customer demand.

This insight helps the business:

- Recognize high-performing restaurant partners
- Prioritize promotions or partnerships
- Support operational planning and revenue growth decisions

The parameterized procedure makes the analysis reusable and scalable, allowing quick ranking for any required Top N value.

INDEXING STRATEGY

– Index on order_date (for monthly reports)

```
create index ind_order_date on orders(order_date);
```

-- Index on customer_name (for joins)

```
create index ind_customer_name on customers(name);
```

-- Index on restaurant_name

```
create index ind_restaurant_name on restaurants(rest_name);
```

1. Business Question
How can we optimize query performance for frequent reports and joins involving orders, customers, and restaurants, especially for monthly analysis and high-volume data?

2. Tables Used

- orders

order_date → Used for monthly and time-based reporting

- customers

name → Used in joins and customer-level analysis

- restaurants

rest_name → Used in joins and restaurant-wise reporting

3. Business Insight Derived
These indexes significantly improve query execution speed by allowing the database to quickly locate relevant records instead of scanning entire tables.
This results in:

- Faster monthly and operational reports
- Improved performance of joins across large datasets
- Better scalability as data volume grows

Overall, this supports efficient analytics, timely decision-making, and smoother system performance.

TRIGGERS IMPLEMENTATION

1. **Auto log high value orders (above 1000)**

```
create table high_value_orders_log (
log_id int primary key auto_increment,
order_id int,
customer_id int,
restaurant_id int,
order_amount decimal(10,2),
log_date datetime default current_timestamp
);
create trigger trg_high_valoue_order
after insert on orders
for each row
begin
   if new.order_amount > 1000 then
      insert into high_value_orders_log (order_id, customer_id,
restaurant_id, order_amount)
      values (new.order_id, new.customer_id, new.restaurant_id,
new.order_amount);
   end if;
end;

insert into orders values (1010, 215, 238, '2024-06-01
12:00:00', 1500.00, 100.00, 'Credit Card', 30);
insert into orders values (1011, 216, 239, '2024-06-02
13:00:00', 8000.00, 50.00, 'Cash', 25);
insert into orders values (1021, 236, 246, '2024-06-03
14:00:00', 2000.00, 20.00, 'UPI', 20);
insert into orders values (1022, 237, 247, '2024-06-04 15:00:00',
1455.00, 30.00, 'Debit Card', 35);
insert into orders values (1023, 238, 248, '2024-06-05
16:00:00', 2120.00, 20.00, 'UPI', 40);

select * from high_value_orders_log;
```

1. Business Question
How can the system automatically track and store high-value orders (orders above ₹1000) at the time they are created, without relying on manual reporting?

2. Tables Used

- orders

Source table where new customer orders are inserted.

- high_value_orders_log

Logging table designed to store details of only high-value transactions, including order, customer, restaurant, amount, and timestamp.

3. Business Insight Derived
This trigger ensures that every high-value order is captured in real time as soon as it is placed.
The logged data can be used to:

- Monitor premium customer behavior
- Identify high-revenue restaurants
- Support fraud checks, audits, and VIP customer programs
- Enable quick reporting without scanning the full orders table

Overall, this provides automated, reliable tracking of critical revenue-impacting orders, improving operational efficiency and business oversight.

“This trigger automatically logs all high-value orders above ₹1000 for real-time revenue tracking, auditing, and premium customer analysis.”

TRIGGERS IMPLEMENTATION

2. **Negative Discount**

```
create trigger trg_negative_discount
before insert on orders
for each row
begin
    if new.discount < 0 then
       set new.discount = 0;
    end if;
end;



insert into orders values (1012, 217, 240, '2024-06-03 14:00:00',
500.00, -20.00, 'UPI', 20);
insert into orders values (1013, 218, 241, '2024-06-04
15:00:00', 600.00, -30.00, 'Debit Card', 35);

select * from orders where order_id = 1012;
```

1. Business Question

How can the system ensure data quality and revenue accuracy by preventing invalid negative discount values from being stored in the orders data?

2. Tables Used

- orders

Contains transactional order details such as:

- order_amount
- discount
- payment_mode
- order_date

A BEFORE INSERT trigger is applied to this table

3. Business Insight Derived

The trigger automatically corrects negative discount values to zero before data is inserted, ensuring:

- Accurate net revenue calculations
- Consistent and clean transactional data
- Protection against data entry or system errors
- Reliable financial and reporting metrics

This logic enforces business rules at the database level, improving overall data integrity without relying on manual validation.

“This trigger ensures discount values remain valid, protecting revenue accuracy and data quality at the source.”

TRIGGERS IMPLEMENTATION

3. **Delivery delay warning**

```
create table delivery_delay_log (
log_id int primary key auto_increment,
order_id int,
customer_id int,
restaurant_id int,
delivery_time int,
created_at timestamp default current_timestamp
);

create trigger log_delivery_delay
after insert on orders
for each row
begin
   if new.delivery_time > 45 then
      insert into delivery_delay_log (order_id, customer_id,
restaurant_id, delivery_time)
      values (new.order_id, new.customer_id, new.restaurant_id,
new.delivery_time);
   end if;
end;

insert into orders values (1016, 231, 241, '2024-06-04
15:00:00', 600.00, 30.00, 'Debit Card', 50);
insert into orders values (1017, 232, 242, '2024-06-05 16:00:00',
700.00, 20.00, 'UPI', 65);
insert into orders values (1018, 233, 243, '2024-06-06
17:00:00', 800.00, 10.00, 'Cash', 48);
insert into orders values (1019, 234, 244, '2024-06-07
18:00:00', 900.00, 0.00, 'Credit Card', 55);
insert into orders values (1020, 235, 245, '2024-06-08
19:00:00', 1000.00, 50.00, 'UPI', 60);

select * from delivery_delay_log;
```

1. Business Question

Which orders are experiencing delivery delays beyond the acceptable threshold (45 minutes), so that delayed deliveries can be tracked, monitored, and acted upon automatically?

2. Tables Used

- orders

Contains order-level details including delivery time for each order.

- delivery_delay_log

A log table that automatically stores details of only delayed orders using a database trigger.

3. Business Insight Derived

This setup enables real-time monitoring of delayed deliveries without manual checks.

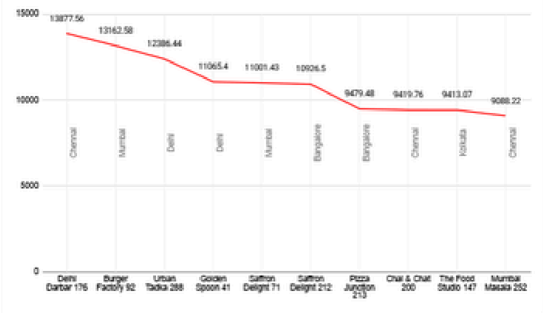
It helps the business:

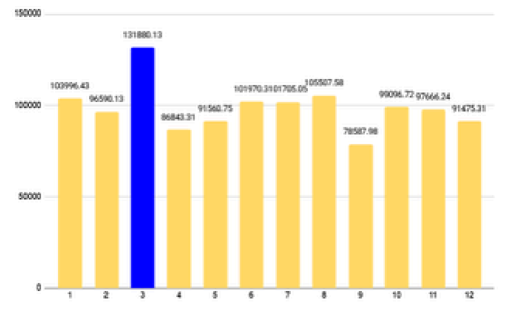
- Identify problematic orders, restaurants, or delivery partners
- Improve delivery SLA compliance
- Support customer service actions like refunds or apologies
- Provide data for operational performance analysis

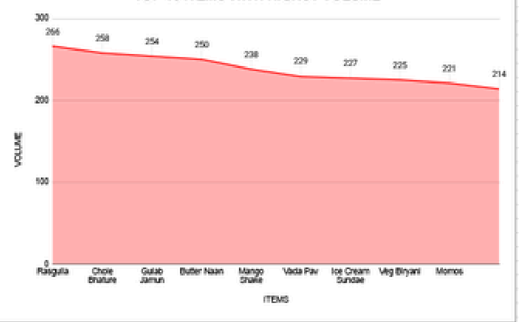
This is an example of automated operational monitoring using triggers, ensuring data quality and faster decision-making.

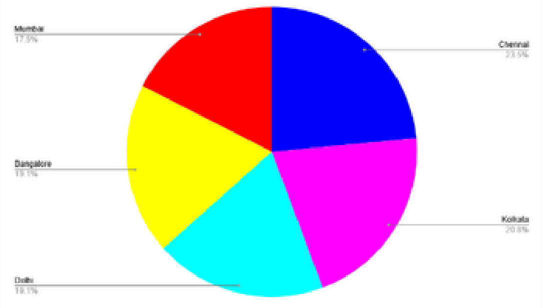
DASHBOARD & VISUALIZATION

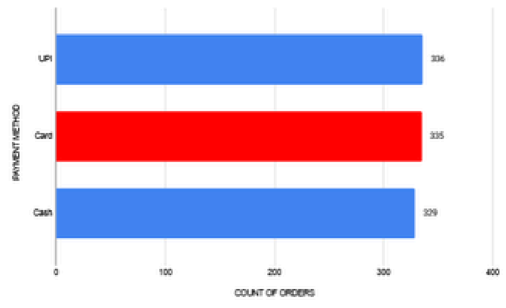
TOTAL REVENUE	TOTAL ORDERS	TOTAL CUSTOMERS	TOTAL RESTAURANTS	AVERAGE ORDER VALUE	AVG DELIVERY TIME
1186879.93	1012	300	300	1192.42	40.2

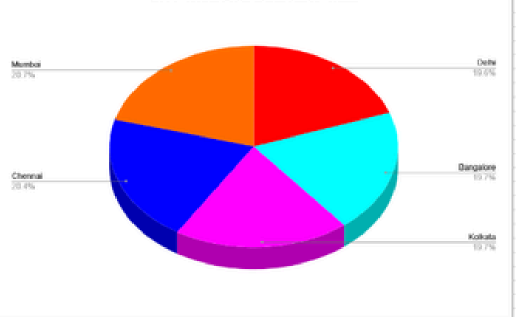

TOP 10 RESTAURANTS BY REVENUE

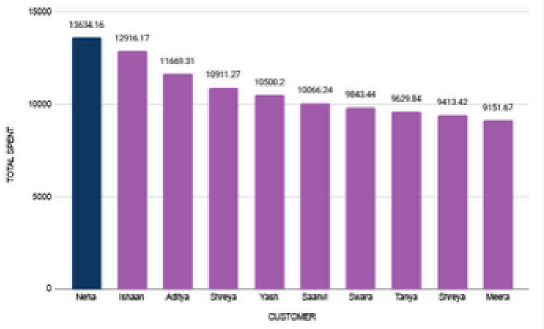

TOTAL REVENUE PER MONTH

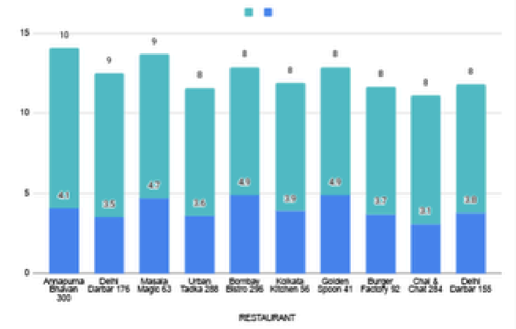

TOP 10 ITEMS WITH HIGHST VOLUME

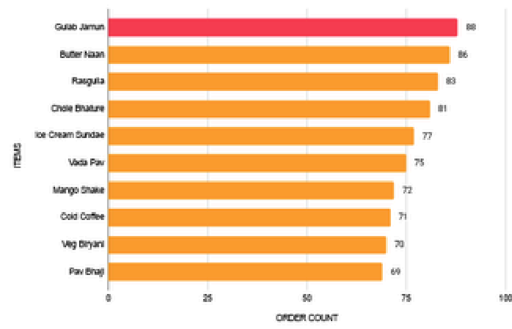

CITY WISE TOTAL ORDERS

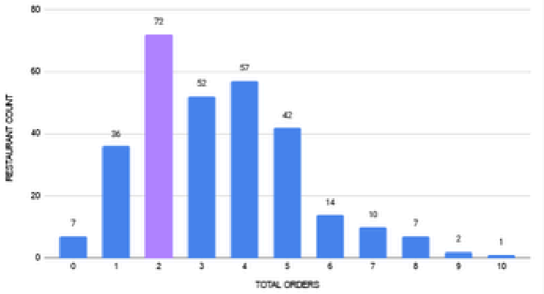

PAYMENT METHOD DISTRIBUTION

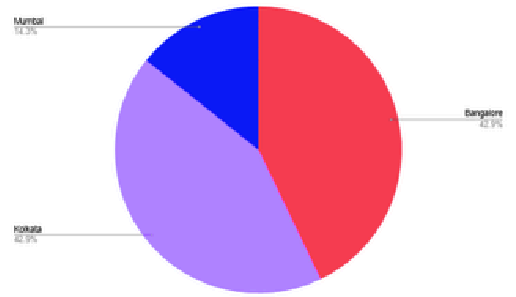

CITY WISE AVG DELIVERY TIME

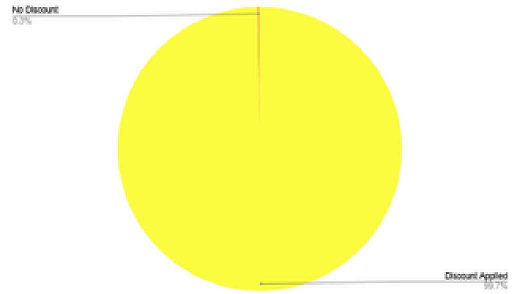

TOP 10 ORDERS BY SPENDING


TOP 10 RESTAURANTS BY ORDER,RATING


TOP 10 MOST FREQUENTLY ORDERED ITEMS


ORDER COUNT DISTRIBUTION


INACTIVE RESTAURANTS BY CITY


DISCOUNT IMPACT ON REVENUE

Overall Business Performance (KPI Overview)

The KPI summary provides a high-level snapshot of the platform’s operational and financial health.

◆ Total Revenue – 118,679.93

The platform has generated strong overall revenue, indicating healthy order volume and consistent customer spending. This reflects effective demand generation and successful monetization through orders.

Business Meaning:

The business is generating steady cash flow and has crossed a meaningful revenue scale.

◆ Total Orders – 1,012

Over a thousand orders have been placed, showing strong customer engagement with the platform.

Business Meaning:

Customers are actively transacting, confirming product-market fit and platform usability.

◆ Total Customers – 300

The platform serves 300 unique customers, indicating a reasonably sized active user base.

Business Meaning:

There is a solid foundation of users, with scope to increase repeat frequency and customer lifetime value.

◆ Total Restaurants – 300

An equal number of restaurants and customers suggests a balanced supply-demand ecosystem.

Business Meaning:

The platform has successfully on boarded restaurants, but growth now depends on improving restaurant utilization rather than adding more listings.

◆ Average Order Value (AOV) – 1,192.42

Customers spend around ₹1,200 per order on average, indicating mid-to-high value transactions.

Business Meaning:

Pricing, menu mix, and discount strategies are effective in maintaining healthy order sizes.

◆ Average Delivery Time – 40.2 minutes

The average delivery time is just over 40 minutes, which is within acceptable industry standards.

Business Meaning:

Delivery operations are stable, but reducing this further could significantly improve customer satisfaction and repeat orders.

Revenue Trends & Distribution

🔹📈 Monthly Revenue Trend

- Major Trends & Patterns

Revenue remains largely stable across the year, with a clear peak in March (₹1,31,880), indicating strong seasonal demand. The lowest revenue is observed in September (₹78,588), suggesting a temporary dip rather than a sustained decline.

- Customer Behavior

Customers show higher spending during peak periods, likely driven by seasonal factors, promotions, or increased ordering frequency. Demand remains consistent in most other months, reflecting habitual platform usage.

- Revenue Distribution

Revenue is evenly distributed across months, with no extreme volatility. Apart from the March peak and September dip, monthly revenues stay within a predictable range, indicating a resilient revenue model.

- Performance Insights

The business benefits from steady baseline demand with identifiable peak opportunities. Replicating successful March strategies during weaker months like September can help smooth revenue and drive incremental growth.

🔹🏆 Top Restaurants by Revenue

- Major Trends & Patterns

Revenue is concentrated among a small set of high-performing restaurants, with the top restaurant generating a noticeably higher share than the rest. Multiple cities—Chennai, Mumbai, Delhi, Bangalore, and Kolkata—appear in the top 10, indicating diversified geographic contribution rather than dependence on a single market.

- Customer Behavior

Customers consistently prefer a few established restaurants, suggesting strong brand trust, repeat ordering behavior, and menu appeal. The presence of the same restaurant brand across different cities highlights customer familiarity and cross-city demand.

- Revenue Distribution

The revenue decline from rank 1 to rank 10 is gradual, not steep, showing a healthy long tail of performers while still reflecting a Pareto-style concentration among top contributors.

- Performance Insights

Top-performing restaurants should be prioritized for partnerships and visibility, as they form a stable revenue backbone. Best practices from these restaurants—such as pricing, menu mix, and service efficiency—can be replicated to uplift mid-tier and under performing partners.

🔹Top 10 Restaurants by order and rating

The top restaurants show a clear volume–quality divide. Annapurna Bhavan and Masala Magic lead in order volume, making them key traffic drivers, while Bombay Bistro and Golden Spoon achieve the highest customer ratings (4.9) despite slightly lower order counts, indicating premium customer satisfaction.

- Customer Behavior & Patterns:

Customers prioritize availability and familiarity for frequent ordering, but consistently reward quality with higher ratings. This suggests two segments: high-volume convenience seekers and quality-focused diners.

- Revenue & Performance Insights:

High-volume restaurants anchor revenue, while high-rated niche players represent growth opportunities through increased visibility. Lower-rated yet high-order outlets (e.g., Chai & Chat) pose brand risk and should be targeted for quality improvement.

Management Actions:

- Protect high-volume leaders with operational support
- Boost visibility of high-rated restaurants to scale revenue
- Intervene with low-rated, high-volume outlets to improve service and safeguard platform reputation

👥 Customer Behavior Analysis

🔹Top 10 Orders by Spending

- Major Trends

Spending is led by a small group of customers, with the top three (Neha, Ishaan, Aditya) significantly outperforming others.

- Patterns

A gradual decline in total spend across the top 10 indicates clear customer spending tiers rather than random behavior.

- Customer Behavior

Repeat appearance of Shreya reflects high purchase frequency and strong platform engagement from certain power users.

- Revenue Distribution

Revenue follows a Pareto-style distribution, where a limited number of customers contribute a disproportionate share of total spend.

- Performance Insights

Retaining top spenders and converting mid-tier customers into high-value users offers the highest return on investment.

🔹🏆 Top 10 most frequently ordered items

- Major Trends & Patterns

Customer demand is strongly concentrated around comfort foods and desserts, with items like Gulab Jamun, Butter Naan, Rasgulla, and Chole Bhature consistently topping the list. The narrow gap between ranks indicates steady demand across the top items rather than reliance on a single product.

- Customer Behavior

Customers show a preference for familiar, indulgent, and repeat-purchase items, suggesting habitual ordering behavior. Beverages and desserts frequently appear alongside main dishes, indicating add-on buying tendencies.

- Revenue Distribution

While individual items may be moderately priced, their high order frequency makes them significant revenue contributors. These items likely drive volume and support upselling opportunities.

- Performance Insights

Top-performing items should be used as anchor products for bundles and promotions to increase cart value. Optimizing visibility and availability of these items can sustain order volume and improve overall platform revenue.

🔹🏆Top 10 items with highest volume

- Major Trends & Patterns

The highest-selling items are predominantly comfort foods and quick-consumption dishes such as Rasgulla, Chole Bhature, and Gulab Jamun. Volumes decline gradually across the top 10, indicating consistent demand rather than dependence on a single item.

- Customer Behavior

Customers show a strong preference for familiar, affordable, and repeat order items, suggesting habitual ordering behavior rather than one time experimentation. Desserts and snacks feature prominently, indicating frequent add-on purchases.

- Revenue Distribution

While these items may not all be high-priced, their high order volume makes them key revenue drivers. They likely contribute significantly to total revenue through frequency rather than margin.

- Performance Insights

These “power items” should be prioritized for: Bundling and combo offers, Visibility during peak hours

🌃City-wise Performance Patterns

🔹City wise total orders

Order volumes are well distributed across cities, with Chennai leading at 235 orders, followed by Kolkata (208), Delhi and Bangalore (191 each), and Mumbai (175). The difference between the highest and lowest cities is moderate, indicating balanced demand across regions.

- Key Patterns & Customer Behavior:

Customers across cities show similar ordering frequency, suggesting consistent adoption of the platform. Slightly higher activity in Chennai and Kolkata may be driven by stronger restaurant availability, local food preferences, or better delivery efficiency.

- Revenue & Performance Insights:

The even distribution of orders reduces geographic risk and supports scalable expansion. Cities with slightly lower order volumes, such as Mumbai, represent growth opportunities through targeted promotions and localized marketing.

🔹City wise average delivery time

- Major Trends & Patterns

Average delivery times are tightly clustered between 39–42 minutes across all cities, indicating consistent operational performance. Delhi, Bangalore, and Kolkata show slightly faster deliveries, while Chennai and Mumbai are marginally slower.

- Customer Behavior

Customers across cities experience similar delivery timelines, creating a uniform service expectation and reducing the risk of city-specific dissatisfaction.

- Revenue Distribution Impact

Cities with faster delivery times (Delhi, Bangalore, Kolkata) are better positioned to support higher repeat orders, while slower cities may face constraints on order frequency if delays increase.

- Performance Insights

Overall logistics performance is stable and scalable. Targeted improvements in Chennai and Mumbai such as route optimization or restaurant preparation time can further enhance customer satisfaction and protect revenue growth.

🌆Payment Behavior Patterns

🔹Payment Method Distribution

Orders are almost evenly split across UPI (336), Card (335), and Cash (329), showing no dominant payment preference among customers.

- Major Trend: Digital payments (UPI + Card) slightly lead, showing strong adoption of cashless transactions.
- Customer Behavior: Customers are comfortable switching between payment modes based on convenience and context.
- Revenue Distribution: Revenue risk is diversified, as the platform is not dependent on a single payment channel.
- Performance Insight: Maintaining multiple payment options is critical; pushing digital payments through incentives can further reduce operational costs without affecting order volume.

Restaurant Payment Activity

🔹Order count distribution

Most restaurants receive 2–4 orders, with the peak at 2 orders (72 restaurants). A large number also sit at 3–5 orders, showing moderate but uneven demand. Only a handful of restaurants (1–2) achieve 9–10 orders, while 7 restaurants have zero orders, indicating under performance or inactivity.

- Major Trend: Order volume is unevenly distributed, with most restaurants operating below peak capacity.
- Pattern: A small number of restaurants capture higher demand, while many remain underutilized.
- Customer Behavior: Customers tend to repeatedly order from a limited set of preferred restaurants.
- Revenue Distribution: Revenue is likely concentrated among these high-order restaurants rather than evenly spread.
- Performance Insight: There is significant opportunity to uplift mid- and low-performing restaurants through better visibility, promotions, and quality bench marking from top performers.

🔹🚫 Inactive Restaurants by City

- Major Trends & Patterns

Bangalore and Kolkata show the highest number of low-order or inactive restaurants (3 each), while Mumbai has only 1. This indicates uneven restaurant utilization across cities despite overall platform activity.

- Customer Behavior

Customers tend to concentrate orders among a limited set of popular restaurants, leaving several others underutilized. This suggests preference-driven ordering rather than lack of overall demand.

- Revenue Distribution

Revenue is likely concentrated among fewer high-performing restaurants, increasing dependency on select partners while limiting monetization from inactive listings.

- Performance Insights

Inactive restaurants signal on boarding, visibility, or quality gaps. Addressing these through targeted promotions, ranking adjustments, or performance reviews can improve city-level efficiency and overall revenue contribution.

🏷️ Discount Impact on Revenue

🔹🏷️ Discount vs No Discount

The data shows an overwhelming reliance on discounted orders, with 99.7% of total orders placed under discount offers, contributing almost the entire revenue base. Orders without discounts are negligible in both volume and revenue contribution.

- Major Trends & Patterns:

Discounts clearly drive customer purchasing decisions. The sharp contrast between discounted and non-discounted orders indicates strong price sensitivity and a clear behavioral preference for promotional pricing.

- Customer Behavior:

Customers are highly responsive to discounts, suggesting that incentives are a key trigger for order placement rather than optional add-ons.

- Revenue Distribution:

Revenue is almost entirely concentrated in discounted transactions, highlighting that current growth is volume-driven through promotions.

- Performance Insights:

While discounts are effective in boosting order volume, the business must assess their impact on profitability. Moving toward targeted, data-driven discounting can help maintain demand while improving margin efficiency.

PERFORMANCE OPTIMIZATION

To ensure efficient query execution and scalability, several database performance optimization techniques were applied throughout the analysis.

🔷 Indexing Strategy

Indexing was implemented on frequently filtered and joined columns such as order_id, customer_id, restaurant_id, order_date, and city. This significantly reduced query execution time by minimizing full table scans and enabling faster data retrieval, especially for aggregation and join-heavy queries.

🔷 Query Optimization Techniques

-Used SELECT only required columns instead of SELECT * to reduce I/O overhead.

-Applied WHERE filters early to limit data volume before aggregation.

-Replaced nested subqueries with CTEs where appropriate for better readability and optimizer efficiency.

-Avoided unnecessary calculations inside loops and repeated aggregations.

🔷 Join Efficiency

-Ensured joins were performed on indexed primary and foreign keys, improving join performance.

-Used INNER JOINs instead of LEFT JOINs where unmatched records were not required.

-Maintained proper join order, starting with smaller, filtered datasets to reduce intermediate result sizes.

🔷 CTEs & Temporary Tables

-CTEs were used to simplify complex logic such as customer segmentation and ranking, improving maintainability.

-Temporary tables were leveraged for intermediate aggregations in multi-step analyses, reducing redundant computations and improving overall query performance.

🔷 Scalability Considerations

-Queries were designed to handle growing transaction volumes by relying on indexed access paths and modular query structures.

-Aggregations and rankings were optimized to scale efficiently as data size increases.

-This approach ensures the system remains performant even with higher order volumes, additional cities, and increased customer activity.

KEY FINDINGS SUMMARY

- Strong Revenue Performance with Stable Growth

The platform generated ₹11,86,879 in total revenue with consistent monthly performance, indicating a stable and scalable business model supported by predictable cash flows and sustained customer demand.

- Diversified city level demand

Order volume is well distributed across major cities, with the top city contributing less than one-fourth of total orders. This balanced geographic spread reduces over-dependence on a single market and provides resilience against region-specific demand fluctuations.

- Revenue Concentration Among High-Value Customers

A small segment of repeat, high-spending customers contributes a disproportionately large share of revenue, highlighting the critical role of customer retention, loyalty programs, and personalized engagement.

- Geographic Revenue Concentration with Expansion Potential

Cities such as Kolkata and Bangalore account for a higher share of orders, while other cities lag behind, indicating clear opportunities for targeted city-level growth and market penetration strategies.

- Digital Payments Drive Transaction Volume

UPI emerges as the dominant payment mode, contributing the highest share of revenue, demonstrating strong digital adoption and enabling faster, low-friction transactions with reduced operational costs.

- Product Demand Concentration Drives Volume

A small group of high-demand menu items contributes a significant portion of total orders, indicating clear customer preference patterns. Leveraging these “power items” through bundling and promotions can increase overall order frequency and revenue uplift.

- Seasonality Influences Demand Patterns

Revenue peaks in specific months (e.g., March) indicate seasonal demand behavior, offering opportunities for proactive marketing, capacity planning, and revenue maximization during high-demand periods.

- Discount Strategy Drives Volume but Pressures Margins

Discounts play a significant role in driving order volume, but they also impact net revenue margins, emphasizing the need for targeted, data-driven discount optimization rather than broad-based promotions.

BUSINESS RECOMMENDATIONS

Revenue Growth

- Adopt city-wise revenue optimization strategies by identifying high performing cities and replicating their successful pricing, restaurant mix, and promotional tactics in under performing regions.
- Implement data-driven discount optimization: since discounted orders contribute to nearly all revenue, focus on controlled, targeted discounts rather than blanket offers to boost volume without eroding margins.
- Promote high-margin restaurants and best selling items through featured placements and in-app visibility, especially during peak demand periods.
- Replicate high-performing monthly strategies (e.g., March peak campaigns) during slower months to stabilize revenue fluctuations.

Customer Retention

- Launch a loyalty or rewards program for repeat and high value customers to encourage consistent ordering behavior.
- Introduce personalized, city-level offers based on customer order history and local preferences, increasing relevance and conversion rates.
- Offer subscription-based benefits such as free delivery or priority support to strengthen long-term engagement and reduce churn.

Cost Efficiency

- Closely monitor discount effectiveness and discontinue low-performing campaigns, as excessive discounts with low order lift can negatively impact profitability.
- Encourage UPI and digital payments through small cashback incentives to reduce cash handling costs and reconciliation effort.
- Optimize delivery route allocation using demand density data to minimize fuel usage and delivery time.

Operational Improvement

- Track and analyze delivery delays at the restaurant and city level to proactively address operational bottlenecks.
- Improve restaurant utilization by promoting low-order restaurants through ranking boosts or limited-time visibility.
- Use automated monitoring (logs, triggers, alerts) to identify SLA breaches and take corrective action in real time.
- Introduce performance-based incentives for delivery agents to improve delivery speed and service consistency.

CONCLUSION

The **Online Food Delivery Business Intelligence & Performance Analysis** Using SQL project delivered a comprehensive, end-to-end analytical solution that converted complex transactional data into clear, decision ready business insights. By integrating structured SQL querying, performance optimized joins, and analytical aggregations, the analysis uncovered key patterns across revenue generation, customer ordering behavior, city-wise performance, and operational efficiency.

The findings highlighted critical revenue contributors, identified under performing segments, and exposed gaps between order volume and customer satisfaction enabling targeted interventions rather than broad assumptions. From a technical perspective, the use of optimized queries, efficient joins, and reusable analytical structures ensured the solution remained scalable, maintainable, and performant as data volume grows.

Overall, the project demonstrates how data analytics can bridge the gap between raw data and strategic action, empowering stakeholders to improve profitability, enhance customer retention, and streamline operations. The outcome is a robust analytical framework that supports sustainable growth, informed decision-making, and long-term business resilience.

FUTURE SCOPE

This project provides a strong analytical foundation that can be further expanded to support advanced, automated, and real-time decision-making. The following enhancements outline the future potential of the solution:

- Automation of Data Pipelines

Automate data ingestion, validation, and transformation processes using scheduled jobs or workflow tools to eliminate manual intervention and ensure consistent data freshness.

- Integration with BI & Visualization Tools

Seamless integration with BI platforms such as Power BI or Tableau can enable dynamic dashboards, drill-down analysis, and role-based reporting for business stakeholders.

- Real-Time Analytics Enablement

By integrating streaming data sources and event-based triggers, the system can evolve into a real-time monitoring solution for orders, revenue, and delivery SLAs, enabling faster operational responses.

- Advanced Predictive & Statistical Modeling

Historical data can be leveraged to build forecasting models for demand, revenue trends, and seasonal patterns, supporting proactive planning and resource allocation.

- Machine Learning Integration

ML models can be introduced for customer segmentation, churn prediction, personalized recommendations, and dynamic discount optimization, further enhancing revenue growth and customer retention.

Overall, these enhancements would transform the project from a descriptive analytics solution into a predictive and prescriptive intelligence platform, enabling smarter, faster, and more scalable business decisions

Email : lizadeka44@gmail.com

LinkedIn : https://www.linkedin.com/in/liza-deka-869473369/