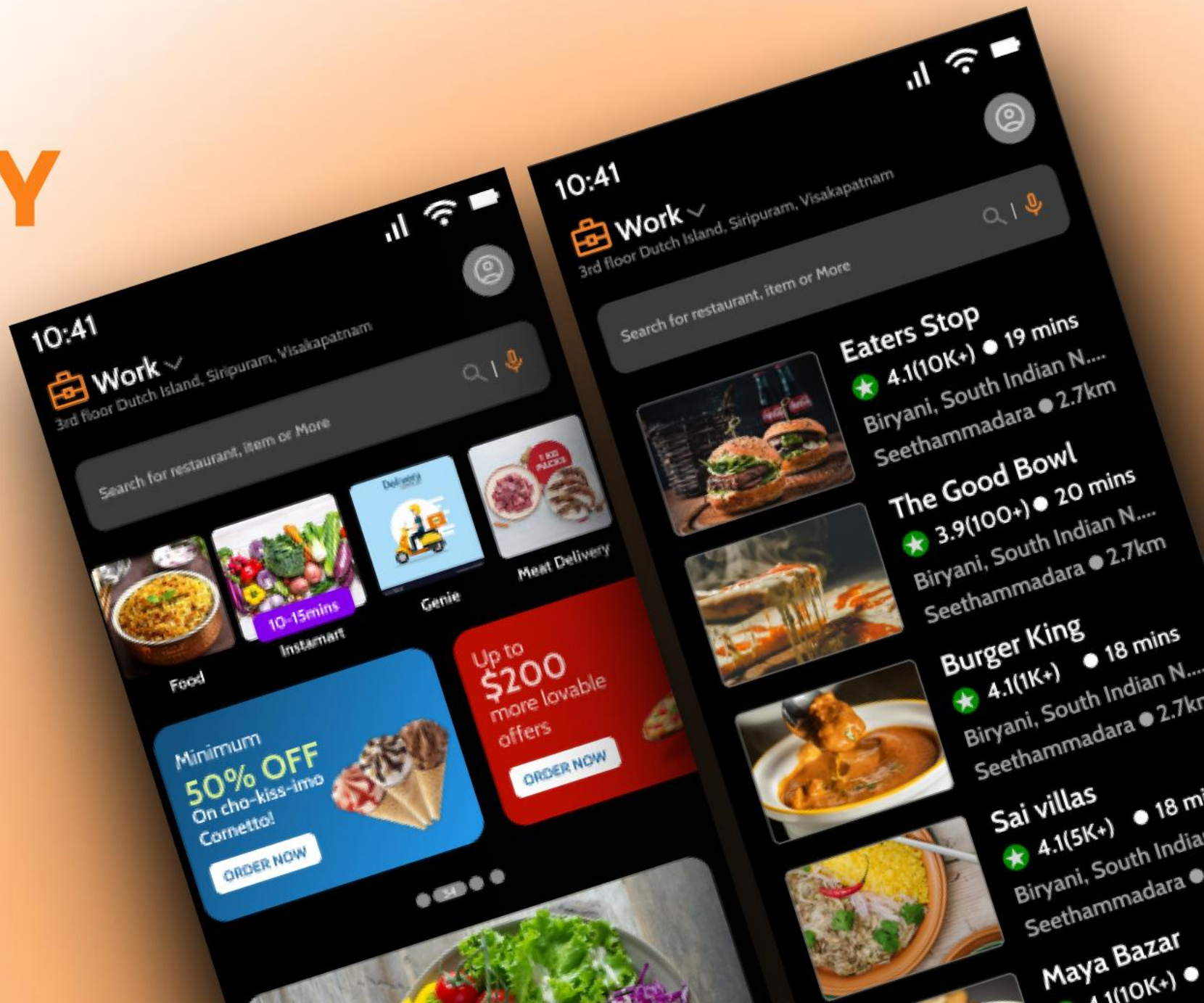




# SQL Data Analysis for Business

By Mahesh





## Swiggy Business Model

**Aggregator platform** - Connects users, restaurants, and delivery partners via an app.

**Diverse Revenue** - Earns from commissions, delivery fees, ads, subscriptions, and Instamart.

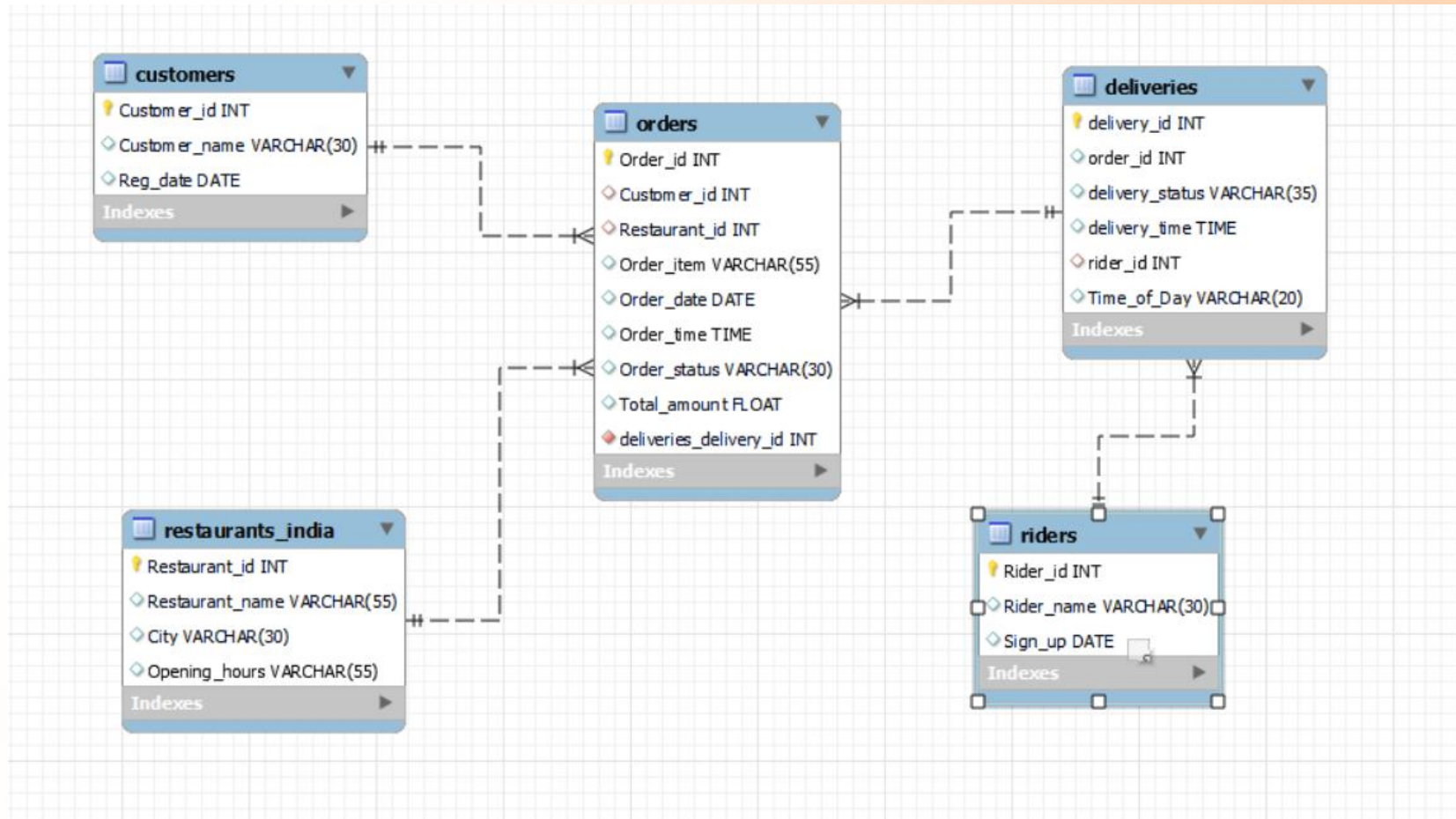
**Strong Logistics** – Own delivery fleet ensures faster and reliable service.

## Project Objective

- Analyze customer behaviour to identify top spenders and repeat purchase items
- Track item and restaurants-wise sales
- Measure revenue performance by restaurant and customer
- Analyze Order Frequency by days , Peak Ordering Timings and Seasonal Trends
- Use SQL-driven insights to support data-backed business decisions



## Database Setup – Snowflake Schema







## Tools Used

- **SQL** – Data modeling, Data exploration, Data Analysis
- **Presentation** – Power Point

# Let's Get into the Business problems



## 1. Popular Time Slots:

Identify the time slots during which the most orders are placed, based on 2-hour intervals.

```
select
> case when hour(order_time) between 0 and 1 then "00:00:00 AM-02:00:00AM"
    when hour(order_time) between 2 and 3 then "02:00:00 AM-04:00:00 AM"
    when hour(order_time) between 4 and 5 then "04:00:00 AM-06:00:00 AM"
    when hour(order_time) between 6 and 7 then "06:00:00 AM-08:00:00 AM"
    when hour(order_time) between 8 and 9 then "08:00:00 AM-10:00:00 AM"
    when hour(order_time) between 10 and 11 then "10:00:00 AM-12:00:00 PM"
    when hour(order_time) between 12 and 13 then "12:00:00 PM-02:00:00 PM"
    when hour(order_time) between 14 and 15 then "02:00:00 PM-04:00:00 PM"
    when hour(order_time) between 16 and 17 then "04:00:00 PM-06:00:00 PM"
    when hour(order_time) between 18 and 19 then "06:00:00 PM-08:00:00 PM"
    when hour(order_time) between 20 and 21 then "08:00:00 PM-10:00:00 PM"
    when hour(order_time) between 22 and 23 then "10:00:00 PM-00:00:00 AM"
end as time_slot,
count(order_id) as Orders_count from orders
group by time_slot
order by Orders_count desc ;
```

	time_slot	Orders_count
►	04:00:00 PM-06:00:00 PM	193
	10:00:00 AM-12:00:00 PM	179
	10:00:00 PM-00:00:00 AM	176
	00:00:00 AM-02:00:00AM	170
	02:00:00 PM-04:00:00 PM	169
	12:00:00 PM-02:00:00 PM	166
	06:00:00 AM-08:00:00 AM	165
	06:00:00 PM-08:00:00 PM	160
	08:00:00 PM-10:00:00 PM	158
	04:00:00 AM-06:00:00 AM	158
	08:00:00 AM-10:00:00 AM	155
	02:00:00 AM-04:00:00 AM	151

- The Most Popular time slots are 4pm to 6pm

## 2. High-Value Customers:

List the customers who have spent more than 3000 in total on food orders.

```
select customer_id, customer_name ,  
round(sum(total_amount),2) as Total_order_amt  
from customers  
join orders using (customer_id)  
group by customer_id, customer_name  
having Total_order_amt >3000  
order by Total_order_amt desc ;
```

customer_id	customer_name	Total_order_amt
189	Jeremiah Carter	3703.08
303	Michelle Norman	3573.09
971	Jennifer Russo	3555.99
1187	Linda Owens	3547.72
127	Megan Edwards	3473.24
1332	Michael Boone	3155.03
415	Kyle Page	3026.61

- Customer id 189,303,971,1187,127,1332,415 are the high value customers who spent above 3k order amount



### 3. Most Popular Dish by City

Identify the most popular dish in each city based on the number of orders.

```
select city, Order_item ,count(order_id) as No_of_orders
from restaurants_india
join orders using (Restaurant_id)
group by city,Order_item
order by No_of_orders desc ;
```

city	order_item	No_of_orders
Chennai	Vada Pav	12
Indore	Pani Puri	12
Mumbai	Pani Puri	12
Patna	Fish Fry	12
Ahmedabad	Vada Pav	11
Kanpur	Mutton Curry	11
Kanpur	Masala Dosa	11
Agra	Pav Bhaji	10
Hyderabad	Hyderabadi Biryani	10
Hyderabad	Pav Bhaji	10

- Though each city has its own popular dish, but the love for Pani puri , vada Pav and Fish Fry is most.

## 4. Order Frequency by day:

Analyze order frequency per day of the week and identify the peak day for each restaurant.

```
select restaurants_india.Restaurant_id,Restaurant_name,  
count(order_id) as Total_orders,  
dayname(order_date) as Day_of_week  
from orders join restaurants_india  
on orders.Restaurant_id=restaurants_india.Restaurant_id  
where order_status="Delivered"  
group by restaurants_india.Restaurant_id,Day_of_week  
order by Total_orders desc ;
```

Restaurant_id	Restaurant_name	Total_orders	Day_of_week
897	Royal Grill	2	Sunday
808	Bombay Grill	2	Sunday
392	Spice Express	2	Monday
641	Tandoori Kitchen	2	Tuesday
88	Kebab Express	2	Thursday
1899	Desi Dhaba	2	Saturday
1109	Spice Palace	2	Tuesday
199	Masala Palace	2	Sunday
1700	Tandoori Palace	2	Monday
1337	Mirchi Cafe	1	Thursday

- Sunday has the highest order frequency ,followed by Monday.

## 5. Ranking City by Revenue

Rank each city based on the total revenue for the last year (2024).

```
select city , round(sum(Total_amount),2) as Total_revenue ,  
row_number() over (order by sum(Total_amount) desc ) as city_Ranking  
from restaurants_india  
join orders using (Restaurant_id)  
where year(Order_date)="2024"  
group by city;
```

city	Total_revenue	city_Ranking
Indore	35165.31	1
Kanpur	35047.18	2
Mumbai	32693.24	3
Patna	30825.8	4
Hyderabad	29701.53	5
Pune	28685.03	6
Delhi	27686.91	7
Surat	27587.03	8
Ahmedabad	27180.31	9
Chennai	26677.76	10

- Indore is the top ranked city by Revenue, followed by Kanpur.

## 6. Order Item Popularity:

Track the popularity of specific order items by seasonal demand spikes.

```
select order_item, count(order_id) as Total_orders,  
       case  
         when month(order_date) between 3 and 5 then "Spring"  
         when month(order_date) between 6 and 8 then "Summer"  
         when month(Ordere_date) between 9 and 11 then "Autumn"  
         else "Winter"  
       end as Seasons  
from orders  
where order_status="Delivered"  
group by order_item, Seasons  
order by Total_orders desc;
```

order_item	Total_orders	Seasons
Dal Makhani	14	Spring
Butter Naan	12	Winter
Gulab Jamun	11	Summer
Paneer Butter Masala	10	Winter
Dal Makhani	10	Autumn
Tandoori Chicken	10	Spring
Fish Fry	10	Summer
Aloo Paratha	10	Winter
Rogan Josh	9	Autumn
Samosa	9	Summer
Tandoori Chicken	9	Winter
Vada Pav	9	Summer
Pav Bhaji	9	Winter

- Winter is the most popular season with higher no of dishes are most selling

## 7. Restaurant Revenue Ranking:

Rank restaurants by their total revenue from the last 1 year within their city.

```
select Restaurant_id , Restaurant_name , City ,  
round(sum(Total_amount),2) as Revenue ,  
dense_rank() over (partition by city  
order by sum(Total_amount) desc )as Ranking  
from restaurants_india  
join orders using (Restaurant_id)  
where year(order_date)=year(current_date())-1  
group by Restaurant_id , Restaurant_name ,City;
```

Restaurant_id	Restaurant_name	City	Revenue	Ranking
1411	Zaika Palace	Indore	2588.59	1
657	Masala Dhaba	Surat	2504.17	1
566	Spice Palace	Lucknow	2214.79	1
167	Desi Corner	Hyderabad	2184.21	1
1431	Bombay Dhaba	Mumbai	2126.11	1
495	Desi Corner	Jaipur	2061.12	1
479	Mirchi Kitchen	Ahmedabad	2049.4	1
1404	Masala Express	Pune	2024.84	1
1699	Bombay Dhaba	Bhopal	1964.11	1
35	Tandoori Grill	Pune	1940.74	2

- Restaurant id 1411 in Indore, 657 in Surat, 566 in Lucknow, 167 in Hyderabad, 1431 in Mumbai are the top ranked restaurants by revenue .



## 8. Order Value Analysis:

Find the average order value (AOV) per customer who has placed more than 5 orders

```
select customer_id ,customer_name ,  
round(avg(Total_amount),2) as AOV ,  
count(order_id) as total_orders from orders  
join customers using (customer_id)  
group by customer_id,customer_name  
having total_orders >5;
```

customer_id	customer_name	AOV	total_orders
1187	Linda Owens	591.29	6
1869	Robert Hood	419.94	6

- Customer id 1187 and 1869 placed over 5 orders each with an average order value of ₹400–₹600.

## 9.Stored Procedure

create a stored procedure to show the full order history of a customer sorted by most recent-- return customer name, order id, restaurant name, order date, Total amount

```
delimiter //  
create procedure customer_details(in customer_id int )  
begin  
select Customer_name, order_id, Restaurant_name,  
Order_item, order_date,Total_amount  
from customers c  
join orders using (Customer_id)  
join restaurants_india using (Restaurant_id)  
where c.Customer_id = customer_id  
order by order_date desc ;  
end //  
delimiter //
```

```
call customer_details(1187);
```

Output :

Customer_name	order_id	Restaurant_name	Order_item	order_date	Total_amoi
Linda Owens	1148	Spice Palace	Chicken Biryani	2025-06-03	526.79
Linda Owens	86	Tandoori Palace	Tandoori Chicken	2024-11-10	650.09
Linda Owens	941	Mirchi Express	Samosa	2024-10-18	567.52
Linda Owens	913	Royal Corner	Hyderabadi Biry...	2024-10-16	569.64
Linda Owens	256	Masala Corner	Masala Dosa	2024-09-05	238.06
Linda Owens	191	Zaika Kitchen	Aloo Paratha	2024-09-01	995.62

## Top Key Insights

- **Peak Ordering Time:** Most orders are placed between 4 PM– 6 PM, while 2 AM – 4 AM sees minimal activity.
- **Top Customers:** Customer id 1187,1869 placed over 5 orders each with an average order value of ₹400–₹600.
- **Seasonal Trends:** Winter has the highest dish sales, while Spring has the lowest, indicating seasonal food demand.
- **Top-Selling Items by City:** Dishes like vada Pav ,Pani puri , Fish Fry , Mutton curry and Masala Dosa are best-sellers across major cities.
- **High Non-Deliveries:** Restaurant id 573,1690,1884 show the highest non-delivery rates.

## Actionable Recommendations

- **Introduce Late-Night Offers:** Encourage more orders during **2 AM – 4AM** with special discounts or combos.
- **Reward Loyal Customers:** Launch a **VIP program** or exclusive deals for high-frequency users to boost retention
- **Plan Seasonal Campaigns:** Launch **season-specific menus** or offers during **Winter**, and create **Spring** discounts to balance demand.
- **Promote Best-Selling Dishes:** Use regional best-sellers in **targeted ads** and **bundled offers** to drive repeat sales.
- **Improve Restaurant Reliability:** Audit and support restaurants with **high non-delivery rates** to reduce cancellations.

Thank You

