SQL Project

Mayank's Kitchen

Analysis of a Food Delivery App Database

PROJECT SUMMARY

01 Objective

To extract actionable insights that can help enhance customer experience, optimize restaurant operations, and drive business growth.

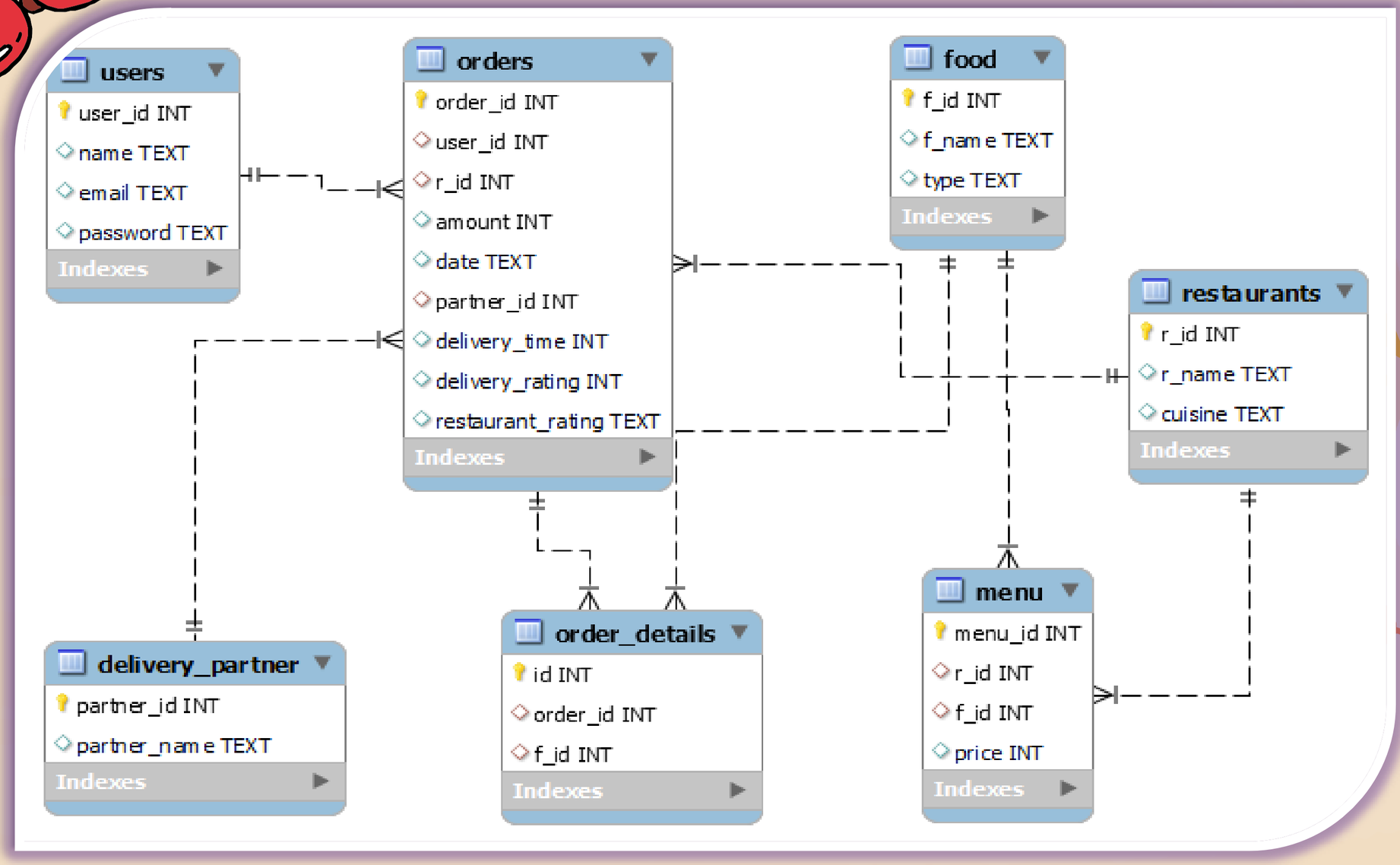
02 About the Dataset

This project focuses on leveraging SQL to perform comprehensive data analysis on a simulated Zomato database.
Check the next slide for Database Schema.

03 Features of the Project

CTE, Window Functions, Joins, Sub Query.

Database Schema



BASIC QUESTIONS

Find the list of customers who have never ordered.

```
SELECT
    *
FROM
    users
WHERE
    user_id NOT IN (SELECT DISTINCT
            (user_id)
        FROM
            orders);
```

Result Grid | Filter Rows: | Export:

user_id	name	email	password
6	Anupama	anupama@gmail.com	46rdw2
7	Rishabh	rishabh@gmail.com	4sw123

Find the avg rating for each partner ID and restaurant.

```sql
SELECT
    partner_name,
    ROUND(AVG(delivery_time), 0) AS avg_del_time,
    ROUND(AVG(delivery_rating), 1) AS avg_partner_rating
FROM
    orders o
        JOIN
    delivery_partner dp ON o.partner_id = dp.partner_id
GROUP BY partner_name order by avg_partner_rating desc;
```

Result Grid | Filter Rows:

partner_name	avg_del_time	avg_partner_rating
Lokesh	35	4.0
Gyandeep	29	3.5
Kartik	42	3.0
Amit	40	3.0
Suresh	46	2.9

```sql
SELECT
    r_name, ROUND(AVG(restaurant_rating), 1) AS avg_rating
FROM
    orders
        JOIN
    restaurants r ON orders.r_id = r.r_id
GROUP BY r.r_name
ORDER BY avg_rating DESC;
```

Result Grid | Filter Rows:

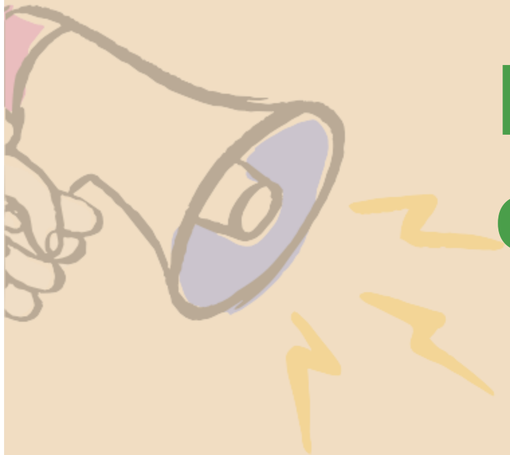
r_name	avg_rating
China Town	3.7
box8	3.5
Dosa Plaza	2.2
kfc	1.4
dominos	1

The same food items may be served at different restaurant. Find the avg price/dish.

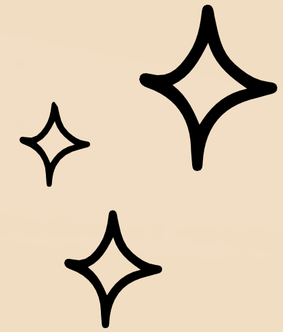
```
 SELECT
    f_name, round(AVG(price),2) AS 'Averag Price'
FROM
    menu
        JOIN
    food ON food.f_id = menu.f_id
GROUP BY f_name
ORDER BY f_name ASC;
```

Result Grid | Filter Rows:

f_name	Averag Price
Chicken Popcorn	300.00
Chicken Wings	230.00
Choco Lava cake	98.33
Masala Dosa	180.00
Non-veg Pizza	450.00
Rava Idli	120.00
Rice Meal	213.33
Roti meal	140.00
Schezwan Noodles	220.00
Veg Manchurian	180.00
Veg Pizza	400.00



Find top 10 restaurants in terms of no. of order for a given month.



```sql
SELECT
    r_name, COUNT(*) AS order_count
FROM
    orders
        JOIN
    restaurants ON orders.r_id = restaurants.r_id
WHERE
    MONTHNAME(date) = 'June'
GROUP BY r_name
ORDER BY order_count DESC
LIMIT 10;    -- Month can be replaced by May/June/July as needed
```

Result Grid | Filter Ro

r_name	order_count
kfc	3
box8	2
dominos	1
Dosa Plaza	1
China Town	1

Restaurants with monthly sales > X.

```sql
SELECT
    r_name, SUM(amount) AS sales
FROM
    orders
        JOIN
    restaurants ON orders.r_id = restaurants.r_id
WHERE
    MONTHNAME(date) = 'June'     -- month
GROUP BY r_name
HAVING sales > 400               -- X value
ORDER BY SUM(amount) DESC
LIMIT 10; -- (Month can be May/June/July and X can be variable)
```

Result Grid

r_name	sales
kfc	990
dominos	950
box8	480

Show all orders from a particular customer in a particular date range.

```sql
SELECT
    name, date, od.order_id, r_name, f_name, amount
FROM
    orders o
        JOIN
    users ON users.user_id = o.user_id
        JOIN
    restaurants r ON r.r_id = o.r_id
        JOIN
    order_details od ON od.order_id = o.order_id
        JOIN
    food ON food.f_id = od.f_id
WHERE
    name = 'Ankit'                                  -- name can be changed
        AND date BETWEEN '2022-05-10' AND '2022-06-10'      -- date range can be set as needed
ORDER BY date ASC;
```

Result Grid | Filter Rows: | Export: | Wrap C

name	date	order_id	r_name	f_name	amount
Ankit	2022-05-15	1016	Dosa Plaza	Masala Dosa	300
Ankit	2022-05-15	1016	Dosa Plaza	Rava Idli	300
Ankit	2022-05-30	1017	Dosa Plaza	Masala Dosa	300
Ankit	2022-05-30	1017	Dosa Plaza	Rava Idli	300

ADVANCED QUESTIONS

Find restaurants with max repeated customers (loyal customers)

```sql
SELECT
    r_name, COUNT(r_name) AS rep_count
FROM
    (SELECT
        r_name, user_id, COUNT(*) AS visits
    FROM
        orders
    JOIN restaurants r ON r.r_id = orders.r_id
    GROUP BY r_name , user_id
    HAVING visits > 1) AS a
GROUP BY r_name
ORDER BY rep_count DESC
LIMIT 1;
```

Result Grid | Filter Rows:

r_name	rep_count
kfc	2
box8	1
Dosa Plaza	1
China Town	1
dominos	1

Month over month revenue growth of Mayank's Kitchen.

```
-- WINDOW Function LAG()
select month, CM_sales, LM_sales,
    (CM_sales-LM_sales)*100/LM_sales as growth
    from (select monthname(date) as month , sum(amount) as CM_sales,
    lag(sum(amount)) over() as LM_sales
    from orders
    group by month) as t;
```

Result Grid | Filter Rows:

month	CM_sales	LM_sales	growth
May	2425	NULL	NULL
June	3220	2425	32.7835
July	4845	3220	50.4658

Find the top 2 spenders for all the 3 months.

```sql
-- Find the top 2 spenders for all the 3 momths
select *
from (select monthname(date) as Month,user_id,sum(amount) as spent ,
rank() over(partition by monthname(date) order by sum(amount) desc)
as Month_rank
from orders group by Month,user_id) as a
where Month_rank < 3
order by Month desc, Month_rank asc;
```

Result Grid | Filter Rows:

Month	user_id	spent	Month_rank
May	1	965	1
May	3	860	2
June	2	1480	1
June	4	800	2
July	5	3035	1
July	2	1190	2

Find customer -wise favorite food.

```sql
-- 1st method (without Common Table Expressions (CTEs))
select customer,food from
(select customer, food, rank() over(partition by customer
order by class desc) as r
from(select users.name as customer, food.f_name as food, count(*) as class
from orders
join order_details on orders.order_id= order_details.order_id
join users on users.user_id= orders.user_id
join food on food.f_id = order_details.f_id
group by users.name, food.f_name
order by users.name) as t) as tt
where r=1
order by customer;
```

Result Grid | Filter Rows:

customer	food
Ankit	Schezwan Noodles
Ankit	Veg Manchurian
Khushboo	Choco Lava cake
Neha	Choco Lava cake
Nitish	Choco Lava cake
Vartika	Chicken Wings

```sql
-- 2nd method using Common Table Expressions (CTEs)
with temp as (
select users.name,f_name,count(*) as frequency
    from orders o join order_details od
    on o.order_id= od.order_id
    join users on users.user_id=o.user_id
    join food on food.f_id=od.f_id
    group by users.name,f_name)

select name,f_name from temp t1
where t1.frequency = (select max(frequency)
from temp t2 where t1.name=t2.name);
```

Find most loyal customers for all restaurants.

```sql
create view v1 as
(select r_name,name, count(*) as visits
from orders o join restaurants r on o.r_id=r.r_id
join users on users.user_id=o.user_id
group by r_name, name
order by r_name);
```

Result Grid | Filter

r_name	name
box8	Nitish
China Town	Ankit
dominos	Neha
Dosa Plaza	Ankit
kfc	Neha
kfc	Vartika

```sql
-- 1st method: without window function
select r_name,name
from v1 t1 where t1 .visits =
(select max(visits) from v1 t2
where t1.r_name =t2.r_name);
```

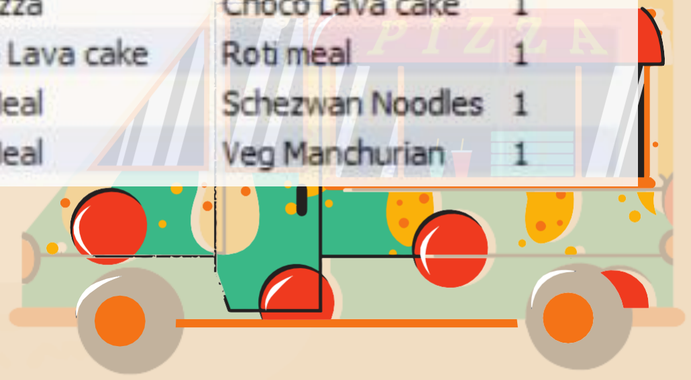
```sql
-- 2nd method: with window fucnction
select r_name,name from
(select *,rank() over(partition by r_name order by visits desc)
as top_visit from v1) as a
where top_visit=1;
```

Find the most paired products (food items ordered together).

```sql
create view v2 as
select orders.order_id,f_id
from orders join order_details od on orders.order_id= od.order_id ;

-- self join to create pairs
create view v3 as
select t1.order_id as order_id,
    t1.f_id as item_1,t2.f_id as item_2
from v2 t1 join v2 t2 on t1.order_id =t2.order_id and t1.f_id<t2.f_id;

create view v4 as
select item_1,item_2, count(*) as freq
    from v3 group by item_1,item_2 order by freq desc;

select f1.f_name as prod_1, f2.f_name as prod_2, freq
from v4 join food f1 on f1.f_id =v4.item_1
    join food f2 on f2.f_id =v4.item_2 order by freq desc;
```

Result Grid | Filter Rows:

prod_1	prod_2	freq
Choco Lava cake	Chicken Wings	5
Non-veg Pizza	Choco Lava cake	4
Schezwan Noodles	Veg Manchurian	4
Chicken Wings	Chicken Popcorn	3
Choco Lava cake	Chicken Popcorn	3
Choco Lava cake	Rice Meal	3
Masala Dosa	Rava Idli	3
Non-veg Pizza	Veg Pizza	1
Veg Pizza	Choco Lava cake	1
Choco Lava cake	Roti meal	1
Rice Meal	Schezwan Noodles	1
Rice Meal	Veg Manchurian	1



THANKS!

Do you have any questions?

- https://www.linkedin.com/in/mayank-srivastava-6a84211054
- https://github.com/Mayank4694
- mayanksri461994@gmail.com