SWIGGY CASE STUDY

GITHUB ACCOUNT:
HTTPS://GITHUB.COM/MUKUND-SHUKLA/SWIGGY--CASE-STUDY

- SQL PROJECT

PROJECT GOAL

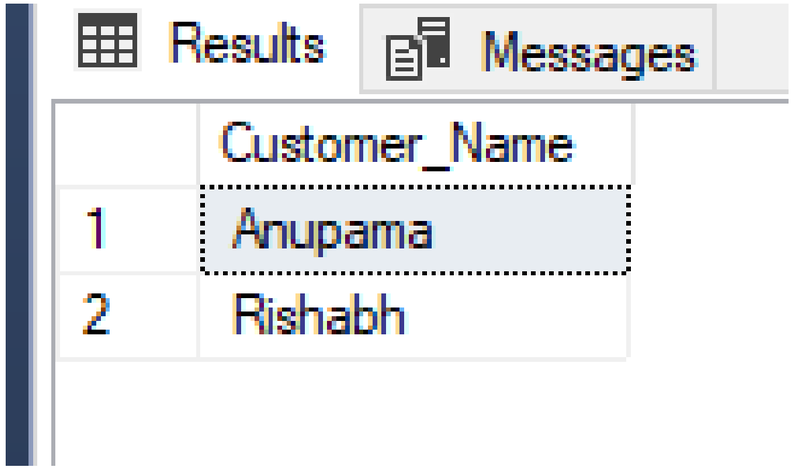
THE MAIN OBJECTIVE OF THIS PROJECT IS TO ANSWER IMPORTANT QUESTIONS FROM THE DATASET TO GAIN A COMPREHENSION OF RESTAURANT SALES REGARDING DIFFERENT CUSTOMER TYPES, THIS CASE STUDY IS SIGNIFICANT FOR ANSWERING IMPORTANT QUESTIONS FOR IMPROVING STRATEGIES, AND ENHANCING OVERALL BUSINESS OPERATIONS .

TABLES

- USER TABLE
- ORDERS TABLE
- FOOD TABLE
- MENU TABLE
- DELIVERY PARTNER TABLE
- RESTAURANT TABLE
- ORDER DETAILS

1. FIND CUSTOMERS WHO HAVE NEVER ORDERED

```
Select u.name as Customer_Name from users$ u where u.user_id
NOT IN (Select o.user_id from orders$ o WHERE o.user_id is not NULL)
```



	Customer_Name
1	Anupama
2	Rishabh

2. WHAT IS THE AVERAGE PRICE PER DISH

```
select food.f_name as food_name,round(AVG(menu.price),0) As AvgPricePerDish
from food INNER JOIN menu On food.f_id=menu.f_id
Group by food.f_name
order by AvgPricePerDish DESC
```

Results | Messages

	food_name	AvgPricePerDish
1	Non-veg Pizza	450
2	Veg Pizza	400
3	Chicken Popcorn	300
4	Chicken Wings	230
5	Schezwan Noodles	220
6	Rice Meal	213
7	Veg Manchurian	180
8	Masala Dosa	180
9	Roti meal	140
10	Rava Idli	120
11	Choco Lava cake	98

3. FIND THE TOP RESTAURANT IN TERMS OF THE NUMBER OF ORDERS FOR A GIVEN MONTH

```
with cte as(
select o.r_id,DATENAME(M,o.date) as Month from orders$ o
),
cte1 as(
select TOP 1 *,COUNT(c.r_id) as Orders from cte c
where MONTH='June'
Group by c.r_id,c.Month
Order by COUNT(c.r_id) DESC
)
select c1.*,c2.r_name Restraunt_Name from cte1 c1 INNER JOIN restaurants$  c2
On c1.r_id=c2.r_id
```

Results | Messages

	r_id	Month	Orders	Restraunt_Name
1	2	June	3	kfc

4. RESTAURANTS WITH MONTHLY SALES GREATER THAN X FOR A PARTICULAR MONTH

```
with cte as(
select o.r_id id,o.amount amt,DATENAME(M,o.date) as Month from orders$ o
)
select r.r_name Restraunt_name,SUM(c.amt) as Monthly_Sales
from cte c INNER JOIN restaurants$ r ON c.id=r.r_id
where MONTH='July'
Group by r.r_name
HAVING SUM(c.amt)>1000
```

Results | Messages

	Restraunt_name	Monthly_Sales
1	China Town	1050
2	dominos	1100
3	kfc	1935

5. SHOW ALL ORDERS WITH ORDER DETAILS FOR A PARTICULAR CUSTOMER IN A PARTICULAR DATE RANGE

```
with cte as(
select * from orders$
where user_id=4 AND date between '2022-06-10' AND '2022-07-10'
),
restraunt as(
select r.r_name,cte.* from cte INNER JOIN restaurants$ r
ON cte.r_id=r.r_id
),
food_ordered as(
select r.*,o.f_id from restraunt r INNER JOIN order_details$ o
on r.order_id=o.order_id
)
select fo.order_id,fo.r_name as Restraunt_name,f.f_name as Food_name
from food_ordered fo INNER JOIN food f On fo.f_id=f.f_id
Group by fo.order_id,fo.r_name,f.f_name
```

Results | Messages

order_id	Restraunt_name	Food_name
1018	Dosa Plaza	Schezwan Noodles
1018	Dosa Plaza	Veg Manchurian
1019	China Town	Schezwan Noodles
1019	China Town	Veg Manchurian

6. FIND RESTAURANT WITH MAX REPEATED CUSTOMERS

```sql
with cte as(
select o.r_id,o.user_id,COUNT(o.user_id) cnt,
ROW_NUMBER() over (partition by o.r_id order by o.r_id ) as rn
from orders$ o
group by o.r_id,o.user_id
HAVING COUNT(o.user_id) >1
)
select TOP 1 cte.r_id Restraunt_Id,r.r_name as Restraunt_Name,
COUNT(cte.user_id) as Repeated_Customers
from cte INNER join restaurants$ r on cte.r_id=r.r_id
Group by cte.r_id,r.r_name
order by COUNT(cte.user_id) desc
```

Restraunt_Id	Restraunt_Name	Repeated_Customers
2	kfc	2

7. MONTH OVER MONTH REVENUE GROWTH OF SWIGGY

```
with sales as(
select SUM(o.amount) as revenue, DATENAME(M,o.date) AS 'MONTH' from orders$ o
Group by DATENAME(M,o.date)
),
growthRate as(
select MONTH,revenue,LAG(revenue,1) over (order by revenue) as GrowthRate from sales
)
select gr.MONTH,gr.revenue,
CONCAT(ROUND((gr.revenue-gr.GrowthRate)/GrowthRate *100,1),'%') as Growth_Rate_Percent
from growthRate gr
order by gr.MONTH desc
```

Results | Messages

	MONTH	revenue	Growth_Rate_Percent
1	May	2425	%
2	June	3220	32.8%
3	July	4845	50.5%

8. FIND THE CUSTOMER'S – FAVORITE FOOD

```
with cte as(
select o.user_id,od.f_id,COUNT(*) as freq,
DENSE_RANK()over(partition by o.user_id order by COUNT(*) desc) as rn
from orders$ o INNER JOIN order_details$ od ON o.order_id=od.order_id
Group by o.user_id,od.f_id
)
select cte.user_id,cte.f_id as food_id,food.f_name as food_name
from cte INNER JOIN food On cte.f_id=food.f_id
where cte.rn=1
```

Results | Messages

	user_id	food_id	food_name
1	1	3	Choco Lava cake
2	2	3	Choco Lava cake
3	3	4	Chicken Wings
4	4	11	Veg Manchurian
5	4	10	Schezwan Noodles
6	5	3	Choco Lava cake

SWIGGY

THANK YOU