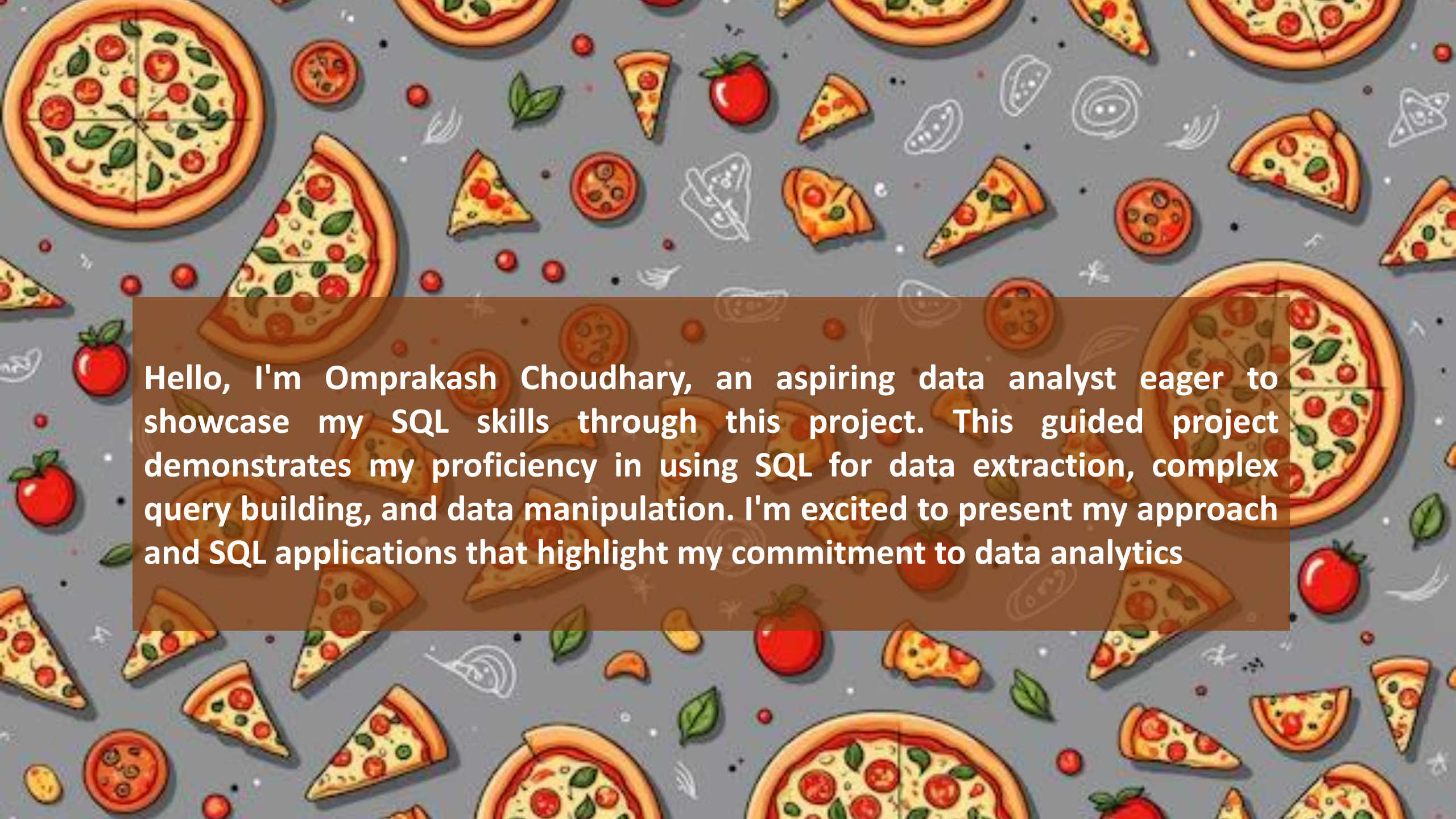




Pizza Sales SQL Project

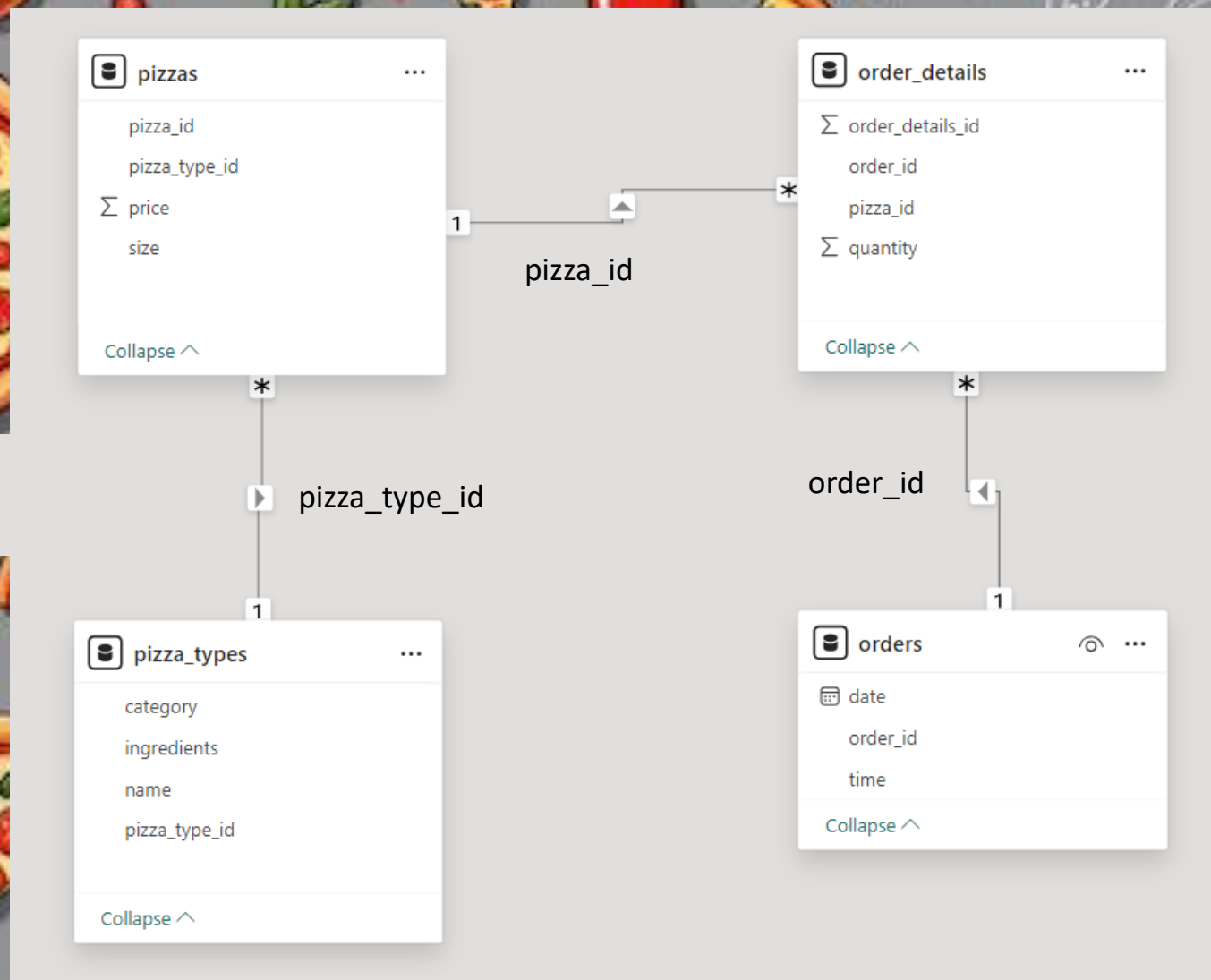
Basic to Advance

By Omprakash Choudhary



Hello, I'm Omprakash Choudhary, an aspiring data analyst eager to showcase my SQL skills through this project. This guided project demonstrates my proficiency in using SQL for data extraction, complex query building, and data manipulation. I'm excited to present my approach and SQL applications that highlight my commitment to data analytics

Data Model



Problem Statement

- Basic:
 - Retrieve the total number of orders placed.
 - Calculate the total revenue generated from pizza sales.
 - Identify the highest-priced pizza.
 - Identify the most common pizza size ordered.
 - List the top 5 most ordered pizza types along with their quantities.
- Intermediate:
 - Join the necessary tables to find the total quantity of each pizza category ordered.
 - Determine the distribution of orders by hour of the day.
 - Join relevant tables to find the category-wise distribution of pizzas.
 - Group the orders by date and calculate the average number of pizzas ordered per day.
 - Determine the top 3 most ordered pizza types based on revenue.
- Advanced:
 - Calculate the percentage contribution of each pizza type to total revenue.
 - Analyze the cumulative revenue generated over time.
 - Determine the top 3 most ordered pizza types based on revenue for each pizza category.

Retrieve the total number of orders placed.

```
SELECT count(order_id) as Total_orders FROM orders;
```

Total_orders
21350

Calculate the total revenue generated from pizza sales.

```
SELECT  
    ROUND(SUM(orders_details.quantity * pizzas.price),  
          2) AS total_sales  
FROM  
    orders_details  
    JOIN  
    pizzas ON orders_details.pizza_id = pizzas.pizza_id;
```

total_sales
817860.05

Identify the highest-priced pizza.

```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

```
SELECT
    pizzas.size, COUNT(orders_details.quantity) AS order_count
FROM
    pizzas
    JOIN
        orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY size
ORDER BY order_count DESC
LIMIT 1;
```

	size	order_count
▶	L	18526

List the top 5 most ordered pizza types along with their quantities.

```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity) AS Most_ordered_pizza
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY name
ORDER BY Most_ordered_pizza DESC
LIMIT 5;
```

	name	Most_ordered_pizza
►	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

Join the necessary tables to find the total quantity of each pizza category ordered.

```
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS Most_ordered_pizza
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY category
ORDER BY Most ordered pizza DESC;
```

	category	Most_ordered_pizza
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS orders
FROM
    orders
GROUP BY HOUR(order_time);
```

	hour	orders
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009
	20	1642
	21	1198
	22	663
	23	28
	10	8
	9	1

Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT
    category, COUNT(name) AS name
FROM
    pizza_types
GROUP BY category;
```

	category	name
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

Group the orders by date and calculate the average number of pizzas ordered per day.

```
SELECT
    ROUND(AVG(total_orders), 0) AS average_orders
FROM
    (SELECT
        orders.order_date AS dates,
        SUM(orders_details.quantity) AS total_orders
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY dates) AS data;
```

	average_orders
▶	138

Determine the top 3 most ordered pizza types based on revenue.

```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

Calculate the percentage contribution of each pizza type to total revenue.

SELECT

```
    pizza_types.category,  
    ROUND((SUM(orders_details.quantity * pizzas.price) / (SELECT  
        SUM(orders_details.quantity * pizzas.price)  
    FROM  
        orders_details  
        JOIN  
        pizzas ON orders_details.pizza_id = pizzas.pizza_id)),  
    4) * 100 AS percentage
```

FROM

```
    pizza_types  
        JOIN  
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
        JOIN  
    orders_details ON orders_details.pizza_id = pizzas.pizza_id  
GROUP BY pizza_types.category  
ORDER BY percentage;
```

	category	percentage
►	Veggie	23.68
	Chicken	23.96
	Supreme	25.46
	Classic	26.91

Analyze the cumulative revenue generated over time.

```
select order_date, round(sum(revenue) over (order by order_date),2) as cum_revenue
from (select orders.order_date, sum(orders_details.quantity * pizzas.price) as revenue
from orders_details join pizzas
on orders_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = orders_details.order_id
group by order_date) as sales;
```

	order_date	cum_revenue
▶	2015-01-01	2713.85
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55
	2015-01-06	14358.5
	2015-01-07	16560.7
	2015-01-08	19399.05
	2015-01-09	21526.4
	2015-01-10	23990.35
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.3
	2015-01-14	32358.7
	2015-01-15	34343.5
	2015-01-16	36937.65
	2015-01-17	39001.75
	2015-01-18	40978.6
	2015-01-19	43365.75

Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
select name, revenue, rn
from
(select category,name, revenue, rank () over(partition by category order by revenue desc) as rn
from
(select pizza_types.category, pizza_types.name, sum((orders_details.quantity)*pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id= pizzas.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where rn <=3;
```

	name	revenue	rn
▶	The Thai Chicken Pizza	43434.25	1
	The Barbecue Chicken Pizza	42768	2
	The California Chicken Pizza	41409.5	3
	The Classic Deluxe Pizza	38180.5	1
	The Hawaiian Pizza	32273.25	2
	The Pepperoni Pizza	30161.75	3
	The Spicy Italian Pizza	34831.25	1
	The Italian Supreme Pizza	33476.75	2
	The Sicilian Pizza	30940.5	3
	The Four Cheese Pizza	32265.70000000065	1
	The Mexicana Pizza	26780.75	2
	The Five Cheese Pizza	26066.5	3



Thank you