



DNYANESHWAR MARKAD

SQL PROJECTION PIZZA SALES





DNYANESHWAR MARKAD

WELCOME TO BRIEF INTRO...

HELLO MY NAME IS DNYANESHWAR, AND IN THIS PROJECT I HAVE UTILIZED SQL QUERIES TO SOLVE QUESTIONS RELATED TO PIZZA SALES.





VISION & MISSION

VISION

- Build a data-driven framework for pizza sales analysis.
- Ensure accurate and efficient decision-making through SQL.
- Convert raw data into actionable business insights.
- Drive business expansion and customer satisfaction.

MISSION

To leverage SQL queries for extracting valuable insights from pizza sales data, enabling data-driven decision-making. By analyzing trends, customer preferences, and sales performance, the goal is to optimize pricing, inventory, and marketing strategies to enhance business growth and profitability.



DNYANESHWAR MARKAD

MEET OUR COMMITTE...



NEIL TRAN
SIMPLE QUESTIONS



CHIDI EZE
INTERMEDIATE QUESTIONS



TEDDY YU
COMPLEX QUESTIONS





NEIL TRAN



Series of question asked
by Neil were....

Simple Questions

- 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.



CHIDI EZE



Series of question asked by Chidi were....

Intermediate Questions

- 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.



TEDDY YU



Series of question asked
by Teddy were....

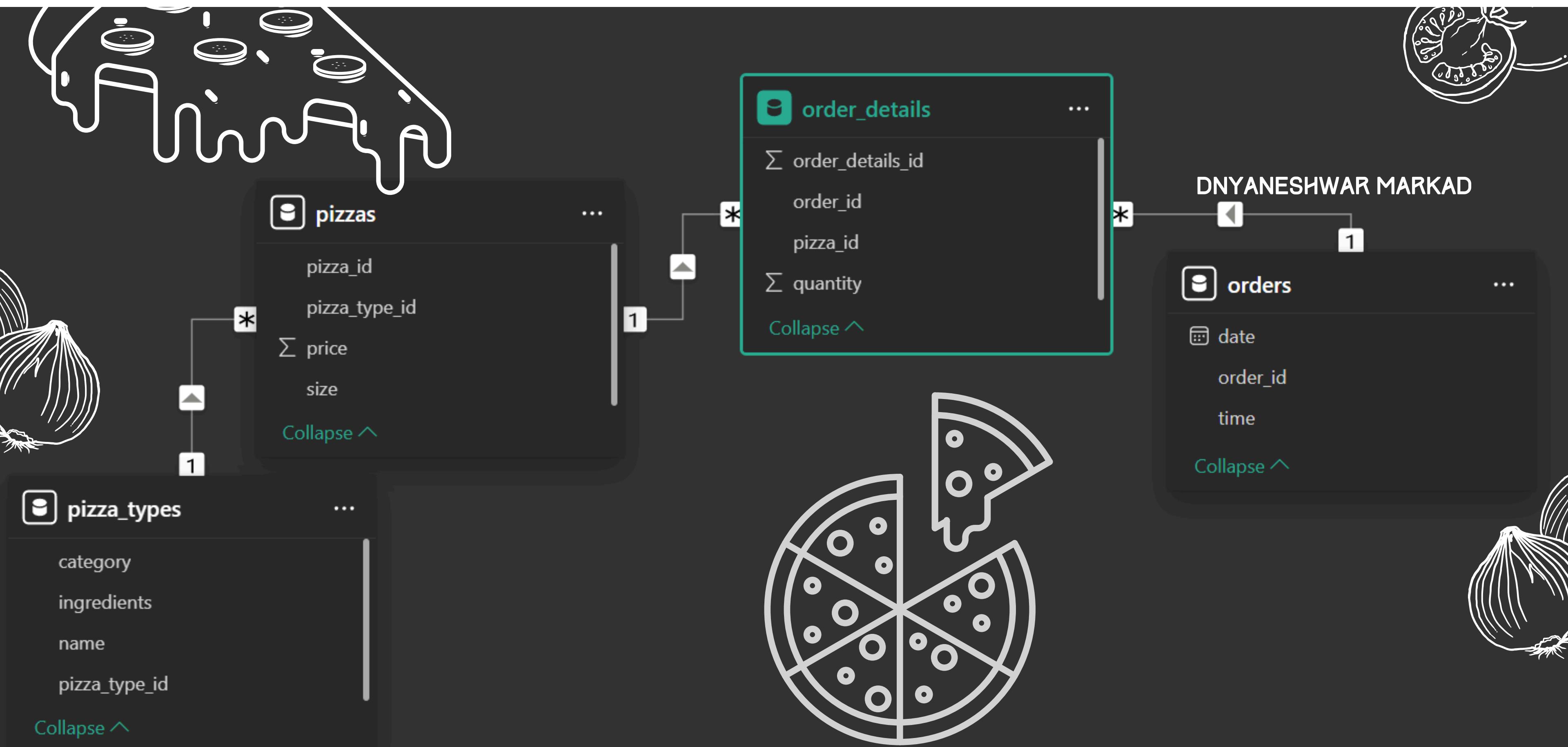
Complex Questions

- 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.

Our Database Schema



DNYANESHWAR MARKAD



Solution using sql queries for the asked questions...

★ Retrieve the total number of orders placed.

```
SELECT  
    COUNT(order_id) AS Total_Ordrs  
FROM  
    orders;
```

	Total_Ordrs
▶	21350

★ Calculate the total revenue generated from pizza sales.

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

	Revenue
▶	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(order_details_id)
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY COUNT(order_details_id) DESC;

```

	size	COUNT(order_details_id)
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

★ Top 5 most ordered pizza types along with their quantities.

```

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

```

	name	no_of_order
▶	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(order_details.quantity)
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.category;
```

	category	SUM(order_details.quantity)
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050

★★ Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(time) AS TIME_of_ORDER, COUNT(order_id)
FROM
    orders
GROUP BY TIME_of_ORDER
ORDER BY TIME_of_ORDER;
```

TIME_of_ORDER	COUNT(order_id)
9	1
10	8
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

★★ Find the category-wise distribution of pizzas.

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

	category	COUNT(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
    orders.date, SUM(order_details.quantity) AS OrderPerDay
FROM
    orders
        JOIN
    order_details ON orders.order_id = order_details.order_id
GROUP BY orders.date;
-- subquery
SELECT
    ROUND(AVG(OrderPerDay), 2)
FROM
    (SELECT
        orders.date, SUM(order_details.quantity) AS OrderPerDay
    FROM
        orders
        JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.date) AS Oder_Quantity;

```

	date	OrderPerDay
▶	2015-01-01	162
	2015-01-02	165
	2015-01-03	158
	2015-01-04	106
	2015-01-05	125
	2015-01-06	147
	2015-01-07	138
	2015-01-08	173
	2015-01-09	127
	2015-01-10	146

	ROUND(AVG(OrderPerDay), 2)
▶	138.47



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

```

SELECT
    pizza_types.name,
    order_details.quantity * pizzas.price AS Revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id;
-- sub query
SELECT
    name, SUM(Revenue)
FROM
    (SELECT
        pizza_types.name,
        order_details.quantity * pizzas.price AS Revenue
    FROM
        pizza_types
        JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN order_details ON order_details.pizza_id = pizzas.pizza_id) AS Pizza_with_Money
GROUP BY name
ORDER BY SUM(Revenue) DESC
LIMIT 3;

```

	name	Revenue
▶	The Hawaiian Pizza	13.25
	The Classic Deluxe Pizza	16
	The Five Cheese Pizza	18.5
	The Italian Supreme Pizza	20.75
	The Mexicana Pizza	16
	The Thai Chicken Pizza	20.75
	The Italian Supreme Pizza	16.5
	The Prosciutto and Arugula Pizza	20.75

	name	SUM(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(pizzas.price * order_details.quantity) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
        2) AS TotalRevenue
    )
    FROM
        order_details
        JOIN
            pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
    2) AS Revenue
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
            order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY category;
```

	category	Revenue
▶	Classic	26.91
	Veggie	23.68
	Supreme	25.46
	Chicken	23.96



★★★ Analyze the cumulative revenue generated over time.

```
select date, sum(Revenue) over(order by date) from
(select orders.date, round(sum(order_details.quantity*pizzas.price),2) as Revenue
from
orders join order_details
on orders.order_id= order_details.order_id
join pizzas
on order_details.pizza_id=pizzas.pizza_id
group by date) as Orders_rev;
```

	date	sum(Revenue) over(order by date)
	2015-02-08	89158.20000000001
	2015-02-09	91353.55000000002
	2015-02-10	93410.05000000002
	2015-02-11	95870.05000000002
	2015-02-12	98028.85000000002
	2015-02-13	100783.35000000002
	2015-02-14	103102.50000000001
	2015-02-15	105243.75000000001
	2015-02-16	107212.55000000002
	2015-02-17	109334.45000000001
	2015-02-18	111977.30000000002
	2015-02-19	114007.55000000002
	2015-02-20	116898.70000000001
	2015-02-21	119009.70000000001
	2015-02-22	120589.65000000001
	2015-02-23	122758.20000000001
	2015-02-24	124952.75000000001
	2015-02-25	127294.05000000002
	2015-02-26	129555.35000000002
	2015-02-27	132413.30000000002
	2015-02-28	134952.90000000002



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

```
select category, Ranking, name, Revenue from
(select category, name, Revenue, rank()over(partition by category order by Revenue desc) as Ranking
from
(select pizza_types.category,pizza_types.name ,round(sum(order_details.quantity*pizzas.price),2) as Revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id= pizzas.pizza_type_id
join order_details
on pizzas.pizza_id= order_details.pizza_id
group by category,name) as Category_Rev) as TopRanked
where Ranking <=3;
```

	category	Ranking	name	Revenue
►	Chicken	1	The Thai Chicken Pizza	43434.25
	Chicken	2	The Barbecue Chicken Pizza	42768
	Chicken	3	The California Chicken Pizza	41409.5
Classic	Classic	1	The Classic Deluxe Pizza	38180.5
	Classic	2	The Hawaiian Pizza	32273.25
	Classic	3	The Pepperoni Pizza	30161.75
Supreme	Supreme	1	The Spicy Italian Pizza	34831.25
	Supreme	2	The Italian Supreme Pizza	33476.75
	Supreme	3	The Sicilian Pizza	30940.5
Veggie	Veggie	1	The Four Cheese Pizza	32265.7
	Veggie	2	The Mexicana Pizza	26780.75
	Veggie	3	The Five Cheese Pizza	26066.5



DNYANESHWAR MARKAD

OUR CONTACT

📞 +91 9960-8248-91

🌐 dnyaneshwar230203@gmail.com

📍 504 shantai Heights, Sinhgad road,
Pune, Maharashtra, India.



DNYANESHWAR MARKAD

THANK YOU!

