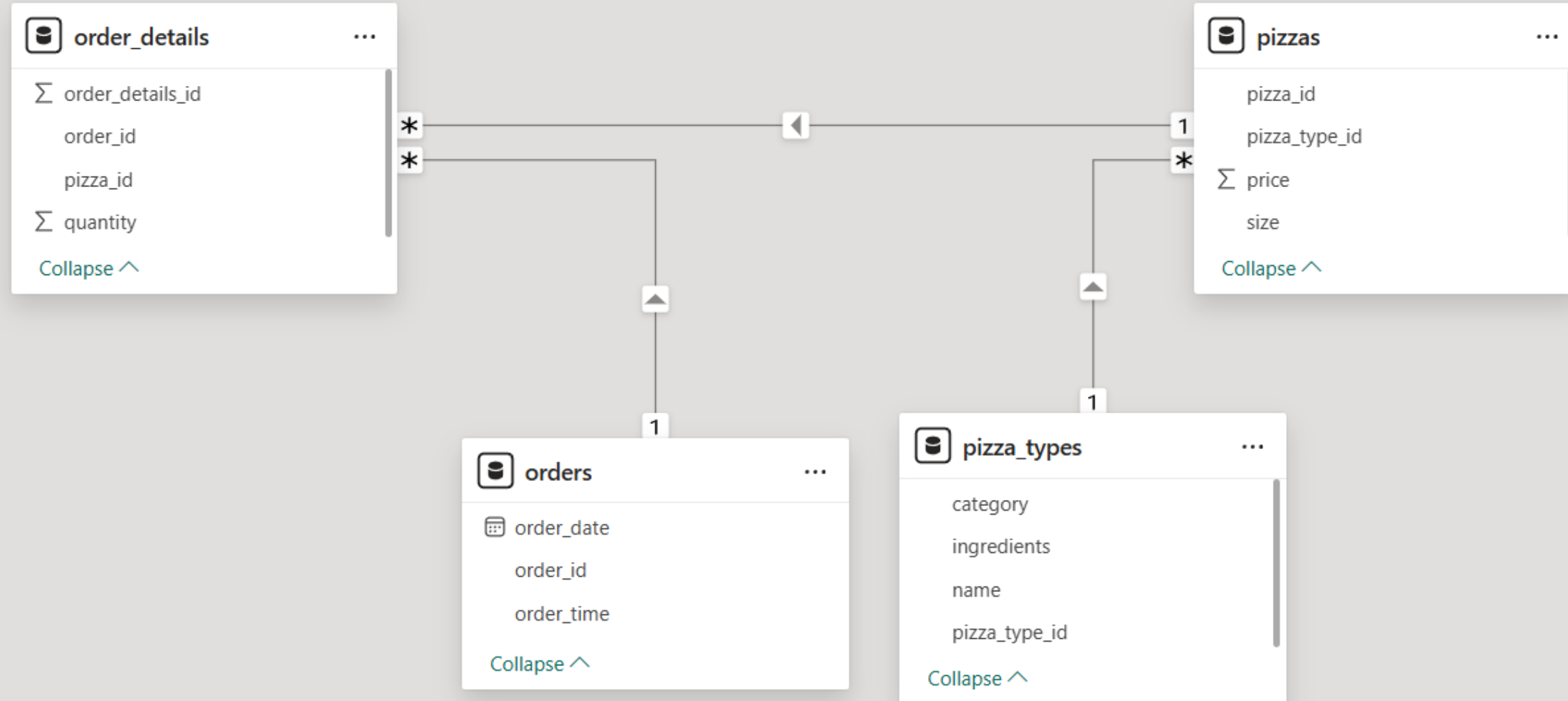# Pizza Sales Analysis: Unveiling Key Insights

In this project, I have utilized SQL queries to solve questions that were related to pizza sales.

by Siddharth Jain

Schema:

Retrieve the total number of orders placed.

21350

Total Orders

Demonstrates a robust and engaged customer base.

Query

```sql
SELECT
    COUNT(order_id) AS Total_Orders
FROM
    orders;
```

The total revenue generated from pizza sales.

$800K+

Total Revenue Generated. Exceeded targets by 10%.

Query

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

Premium Pizzas: Identifying Our Highest-Priced Pizza

The Greek Pizza

Highest priced pizza at $35.95

Query

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

Size Matters: the most common pizza size ordered.

Large (14")

18526 times

Query

```sql
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS Ordered_Times
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY Ordered_Times DESC
LIMIT 1;
```

Top 5 Pizza Types: What Our Customers Crave

Pizza Type	Quantity Ordered
The Classic Deluxe Pizza	2,453
The Barbecue Chicken Pizza	2,432
The Hawaiian Pizza	2,422
The Pepperoni Pizza	2,418
The Thai Chicken Pizza	2,371

Query

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

Category Breakdown: The total quantity of each pizza category ordered.

1 **Classic**
Qty: 14888

2 **Supreme**
Qty: 11987

3 **Veggie**
Qty: 11649

4 **Chicken**
Qty: 11050

Query

```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
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
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

Order Distribution: Optimizing Operations by Hour of Day

1 Morning (9-12 PM)

17.8% of daily orders.

2 Afternoon (1-3 PM)

25.5% of daily orders.

3 Evening (4-7 PM)

40.9% of daily orders.

4 Night (8-11 PM)

15.8% of daily orders.

Query

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

The Category-wise distribution of pizzas

1   **Chicken**
6 types

2   **Classic**
8 types

3   **Supreme**
9 types

4   **Veggie**
9 types

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

The Average number of pizzas ordered per day :

=138

Query

```sql
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

The top 3 most ordered pizza types based on revenue.

The Thai Chicken Pizza
Revenue $ 43434.45

The Barbecue Chicken Pizza
Revenue $ 42768

The California Chicken Pizza
Revenue $ 41409.5

Query

```sql
SELECT
    pizza_types.name,
    SUM(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
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

The percentage contribution of each pizza type to total revenue.

Category Type	Revenue %
Classic	26.91%
Veggie	23.68%
Supreme	25.46%
Chicken	23.96

Query

```sql
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),
                            2) AS Total_Revenue
            FROM
                order_details
                    JOIN
                pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
        2) AS revenue_percentage
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
GROUP BY pizza_types.category;
```

The cumulative revenue generated over time.

$ 817860.05

Total Revenue Generated till date 2015-12-31.

Query

```
select order_date,
sum(revenue) over(order by order_date) as cumulative_revenue
from
(select orders.order_date,
sum(order_details.quantity*pizzas.price) as revenue
from order_details
JOIN pizzas
ON order_details.pizza_id = pizzas.pizza_id
JOIN orders
ON orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

order_date	cumulative_revenue
2015-01-01	2713.8500000000004
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	23990.350000000002