

Pizza Sales Report

Jaydeb Shaw



Objective

The objective of this project was to analyze one year of pizza sales data using SQL. The goal was to find important patterns in customer preferences and sales trends.

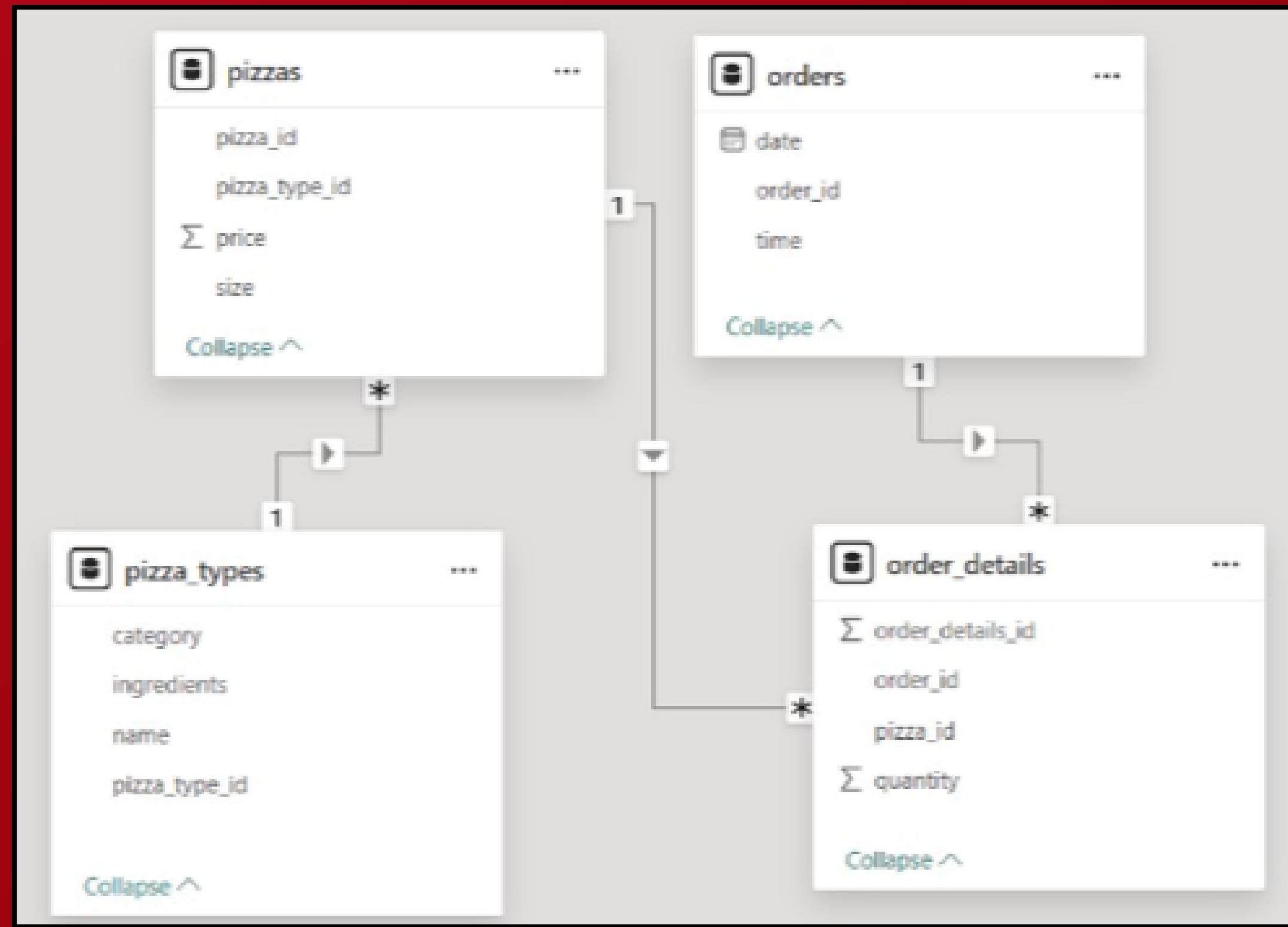
This analysis helps make better decisions for improving inventory management and business operations, which can lead to growth and higher profits in the future.



Problem Statement

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

Data Modelling



Let's Dive
Into the
World of
Pizza Sales
Using



1. Retrieve the total number of orders placed

Query

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

Output

	Total_Orders
▶	21350

2. Calculate the total revenue generated from pizza sales.

Query

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

Output

Total_Revenue_Generated
817860.05

3. Identify the highest-priced pizza.

Query

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

Output

name	price
The Greek Pizza	35.95

4. Identify the most common pizza size ordered.

Query

```
SELECT
    p.size AS Pizza_Size,
    COUNT(o.order_details_id) AS Order_Times
FROM
    pizzas p
        JOIN
    order_details o ON p.pizza_id = o.pizza_id
GROUP BY p.size
ORDER BY COUNT(o.order_details_id) DESC;
```

Output

	Pizza_Size	Order_Times
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

Insight

Out of 21,350 orders, only 540 were for XL pizzas and just 28 for XXL, indicating very low demand. The company might consider discontinuing the XL and XXL size to optimize operations.

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

Query

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

Output

	name	Order_Count
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

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

Query

```
SELECT
    pt.category, SUM(od.quantity) AS Total_Quantity_Ordered
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.category
ORDER BY Total_Quantity_Ordered DESC;
```

Output

	category	Total_Quantity_Ordered
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

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

Query

```
SELECT  
    HOUR(order_time) AS Order_Hour,  
    COUNT(order_id) AS Order_Count  
FROM  
    orders  
GROUP BY Order_Hour  
ORDER BY Order_Count DESC;
```

Output

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

Insight

Sales of pizza pick up strongly from 12 'o' clock. This is probably because people are busy during the day.

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

Query

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

Output

	category	Available_Pizza_Types
▶	Supreme	9
	Veggie	9
	Classic	8
	Chicken	6

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

Query

```
SELECT  
    ROUND(AVG(quantity)) AS average_order_quantity  
FROM  
    (SELECT  
        o.order_date, SUM(od.quantity) AS quantity  
    FROM  
        orders o  
    JOIN order_details od ON od.order_id = o.order_id  
    GROUP BY o.order_date) AS total_orders_per_day;
```

Output

	average_order_quantity
▶	138

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

Query

```
SELECT
    pt.name,
    SUM(od.quantity * p.price) AS Total_Revenue_Generated
FROM
    pizzas p
        JOIN
    pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
        JOIN
    order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.name
ORDER BY Total_Revenue_Generated DESC
LIMIT 3;
```

Output

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

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

Query

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

Output

	category	revenue_percentage
▶	Classic	26.91
	Veggie	23.68
	Supreme	25.46
	Chicken	23.96

12. Analyze the cumulative revenue generated over time.

Query

```
SELECT
    order_date,
    ROUND(SUM(revenue) OVER (ORDER BY order_date)) AS cumulative_revenue
FROM (
    SELECT
        o.order_date,
        SUM(od.quantity * p.price) AS revenue
    FROM
        pizzas p
    JOIN
        order_details od
    ON p.pizza_id = od.pizza_id
    JOIN
        orders o
    ON o.order_id = od.order_id
    GROUP BY
        o.order_date
) AS sales;
```

Output

	order_date	cumulative_revenue
▶	2015-01-01 00:00:00	2714
	2015-01-02 00:00:00	5446
	2015-01-03 00:00:00	8108
	2015-01-04 00:00:00	9864
	2015-01-05 00:00:00	11930
	2015-01-06 00:00:00	14358
	2015-01-07 00:00:00	16561
	2015-01-08 00:00:00	19399
	2015-01-09 00:00:00	21526
	2015-01-10 00:00:00	23990
	2015-01-11 00:00:00	25863
	2015-01-12 00:00:00	27782
	2015-01-13 00:00:00	29831
	2015-01-14 00:00:00	32359

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

Query

```
WITH ranked_pizza AS (
  SELECT pt.name, pt.category, SUM(od.quantity * p.price) AS revenue,
  RANK() OVER (PARTITION BY pt.category ORDER BY SUM(od.quantity * p.price) DESC) AS sales_rank
  FROM pizza_types pt JOIN pizzas p ON pt.pizza_type_id = p.pizza_type_id
  JOIN order_details od ON od.pizza_id = p.pizza_id GROUP BY pt.name, pt.category
)
SELECT category, name, revenue, sales_rank
FROM ranked_pizza
WHERE sales_rank <= 3;
```

Output

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

Key Takeaways

- **Average Orders:** The average number of orders was 138, indicating potential for growth with improved marketing and offers.
- **Focus on Classic Pizzas:** Classic pizzas are customer favorites. Focusing more on them can attract even more customers.
- **Discontinue XL and XXL Sizes:** Sales of XL and XXL pizzas are low. Discontinuing these sizes can reduce costs and streamline the menu.
- **Peak Sales Hours:** Most orders happen between 12 PM and 7 PM. Prioritizing promotions during these hours can maximize sales.
- **Chicken Pizzas Generate High Revenue:** Chicken pizzas bring in the most revenue despite having fewer options. Expanding this category can boost sales.

Thank You!

