



Sales Analysis

Chicago
Pizza

By Suhani

About Dataset

The data for this analysis is sourced from Kaggle's Chicago Pizza Sales Dataset, containing comprehensive sales information from a pizza restaurant chain. The dataset includes:

There are four Tables

- Orders Table: **21,351** rows, 3 columns (Order ID, Order Date, Time).
- Order Details Table: **48,621** rows, 4 columns (Order Details ID, Order ID, Pizza ID, Quantity).
- Pizzas Table: **97** rows, 4 columns (Pizza ID, Pizza Type ID, Size, Price).
- Pizza Types Table: **33** rows, 4 columns (Pizza Type ID, Name, Category, Ingredients).





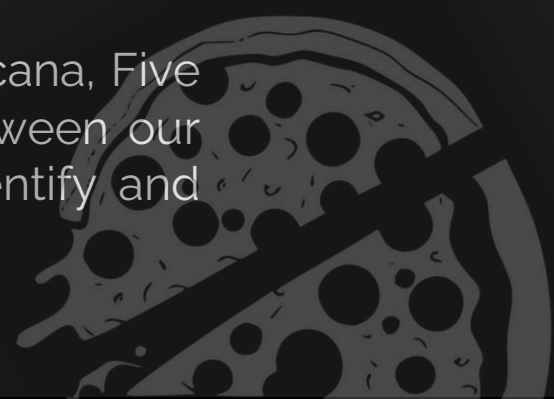
Project Objective

The Pizza Sales Project is a data analysis Project aimed at analyzing sales data from a pizza restaurant chain. Utilizing SQL for data extraction, transformation, and analysis, the project seeks to uncover insights and provide actionable recommendations to optimize menu offerings, enhance marketing strategies, improve operational efficiency, and ultimately boost sales and customer satisfaction.

Business Objective and Problem

Business Objective: The primary goal of the business is to increase sales and profitability by optimizing our pizza menu. This involves enhancing the appeal of underperforming pizzas, leveraging customer preferences, and implementing strategic promotions to attract new customers, retain existing ones, and boost revenue across all categories.

Business Problem: Despite a diverse menu, certain pizzas (California Chicken, Four Cheese, Mexicana, Five Cheese, Spicy Italian, Italian Supreme) are underperforming in sales and revenue. This gap between our offerings and customer preferences results in missed revenue opportunities. Our aim is to identify and develop strategies to stimulate interest and sales of underperformance pizzas.





Retrieve the total number of orders placed.

Solution:

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

Result:

Result Grid		
	Total_Orders	
▶	21350	





Calculate the total revenue generated from pizza sales.

Solution:

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

Result:

Result Grid			
	Total_Revenue		
▶	817860.05		






Identify the highest-priced pizza.

Solution:

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

Result:

Result Grid			Filter Rows:
	name	price	
▶	The Greek Pizza	35.95	





Identify the most common pizza size ordered.

Solution:

```
WITH pizzasize as(  
  SELECT size,  
         COUNT(p.pizza_id) OrderCount,  
         DENSE_RANK() OVER(ORDER BY COUNT(pizza_id) DESC) MostOrderedSize  
  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 size  
)  
SELECT size, OrderCount  
FROM pizzasize  
WHERE MostOrderedSize = 1;
```

Result:

Result Grid			Filter Rows:
	size	OrderCount	
▶	L	18526	





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

Solution:

```
WITH pizza_sold as (  
  SELECT p.pizza_type_id,  
         name,  
         SUM(quantity) quantity,  
         DENSE_RANK() OVER (ORDER BY SUM(quantity) DESC) rnk  
  FROM pizza_types pt  
  JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id  
  JOIN order_details od ON od.pizza_id = p.pizza_id  
  group by p.pizza_type_id, name  
)  
SELECT name, quantity  
FROM pizza_sold  
WHERE rnk <= 5;
```

Result:

Result Grid			Filter Rows:
	name	quantity	
▶	The Classic Deluxe Pizza	2453	
	The Barbecue Chicken Pizza	2432	
	The Hawaiian Pizza	2422	
	The Pepperoni Pizza	2418	
	The Thai Chicken Pizza	2371	





Find the total quantity of each pizza category ordered.

Solution:

```
SELECT
    category, SUM(quantity) quantity
FROM
    pizza_types pt
    LEFT 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 category
ORDER BY quantity DESC;
```

Result:

Result Grid			Filter Rows:
	category	quantity	
▶	Classic	14888	
	Supreme	11987	
	Veggie	11649	
	Chicken	11050	







Determine the distribution of orders by hour of the day.

Solution:

```
SELECT
    HOUR(order_time) hour, COUNT(order_id) order_count
FROM
    orders o
GROUP BY HOUR(order_time)
ORDER BY order_count DESC;
```

Result:

Result Grid   Filter Rows		
	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





Find the category-wise distribution of pizzas.

Solution:

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

Result:

Result Grid			Filter Rows:
	category	pizza_category_count	
▶	Supreme	9	
	Veggie	9	
	Classic	8	
	Chicken	6	





Calculate the average number of pizzas ordered per day.

Solution:

```
SELECT
    ROUND(AVG(Order_By_Date)) avg_pizza_ordered_per_date
FROM
    (SELECT
        order_date, SUM(quantity) Order_By_Date
    FROM
        orders o
    JOIN order_details od ON o.order_id = od.order_id
    GROUP BY order_date
    ORDER BY Order_By_Date DESC) AS qty_ordered;
```

Result:

Result Grid		Filter Rows:
	avg_pizza_ordered_per_date	
▶	138	



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

Solution:

```
WITH pizza_revenue as (  
  SELECT p.pizza_type_id, name, ROUND(SUM(price*quantity)) revenue,  
         DENSE_RANK() OVER(ORDER BY SUM(price*quantity) DESC) revenue_rank  
  FROM pizzas p  
  JOIN order_details od ON p.pizza_id = od.pizza_id  
  JOIN pizza_types pt ON pt.pizza_type_id = p.pizza_type_id  
  GROUP BY p.pizza_type_id, name  
)  
  
SELECT name, revenue  
FROM pizza_revenue  
WHERE revenue_rank <=3;
```

Result:

Result Grid			Filter Rows:
	name	revenue	
▶	The Thai Chicken Pizza	43434	
	The Barbecue Chicken Pizza	42768	
	The California Chicken Pizza	41410	

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

Solution:

```
SELECT
    category,
    CONCAT(percentage_revenue_contri, '%') percentage_revenue_contribution
FROM
    (SELECT
        category,
        ROUND((SUM(price * quantity) / (SELECT
            SUM(price * quantity)
        FROM
            pizzas p
            JOIN order_details od ON od.pizza_id = p.pizza_id)) * 100, 2) AS percentage_revenue_contri
    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 category
    ORDER BY percentage_revenue_contri DESC) percentage_revenue;
```

Result:


Result Grid			Filter Rows:
	category	percentage_revenue_contribution	
▶	Classic	26.91%	
	Supreme	25.46%	
	Chicken	23.96%	
	Veggie	23.68%	

Analyze the cumulative revenue generated over time.

Solution:

```
WITH sales as(
  SELECT order_date, ROUND(SUM(quantity* price), 1) revenue
  FROM pizzas p
  JOIN order_details od ON od.pizza_id = p.pizza_id
  JOIN orders o ON o.order_id = od.order_id
  GROUP BY order_date
)
SELECT order_date,
       revenue,
       ROUND(SUM(revenue) OVER (ORDER BY order_date ROWS UNBOUNDED PRECEDING), 1) cumulative_revenue
FROM sales;
```

Result:

Result Grid			Filter Rows:	
	order_date	revenue	cumulative_revenue	
▶	2015-01-01	2713.9	2713.9	
	2015-01-02	2731.9	5445.8	
	2015-01-03	2662.4	8108.2	
	2015-01-04	1755.5	9863.7	
	2015-01-05	2066	11929.7	
	2015-01-06	2429	14358.7	
	2015-01-07	2202.2	16560.9	
	2015-01-08	2838.3	19399.2	
	2015-01-09	2127.4	21526.6	
	2015-01-10	2464	23990.6	
	2015-01-11	1872.3	25862.9	
	2015-01-12	1919	27781.9	
	2015-01-13	2049.6	29831.5	
	2015-01-14	2527.4	32358.9	
	2015-01-15	1984.8	34343.7	
	2015-01-16	2594.2	36937.9	
	2015-01-17	2064.1	39002	

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

Solution:

```
WITH pizza_ordered as (  
  SELECT category, name, SUM(price* quantity) as revenue,  
         DENSE_RANK() OVER (PARTITION BY category ORDER BY SUM(price* quantity) DESC) pizza_rank  
  FROM pizza_types pt  
  JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id  
  JOIN order_details od ON od.pizza_id = p.pizza_id  
  GROUP BY category, name  
)  
SELECT category, name, revenue  
FROM pizza_ordered  
WHERE pizza_rank <= 3
```

Result:

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



Insights

1. The total number of orders placed in 2015 was 21,350, generating a total revenue of \$817,860.05.
2. The highest-priced pizza was The Greek Pizza, priced at \$35.95.
3. The most commonly ordered pizza size was Large (Size L), with 18,526 orders.
4. Top 5 Most Ordered Pizzas:

The Classic Deluxe Pizza - 2,453 orders
The Barbecue Chicken Pizza - 2,432 orders
The Hawaiian Pizza - 2,422 orders
The Pepperoni Pizza - 2,418 orders
The Thai Chicken Pizza - 2,371 orders

5. Pizza Categories:

Total quantity ordered by category:
Classic: 14,888
Supreme: 11,987
Veggie: 11,649
Chicken: 11,050





Insights

6. Order Distribution by Hour:

Peak order hours were:

12 PM: 2,520 orders

1 PM: 2,455 orders

6 PM: 2,399 orders

5 PM: 2,336 orders

Least busy hours were early mornings and late nights.

7. On average, 138 pizzas were ordered per day.

8. Cumulative Revenue Growth:

The cumulative revenue showed a steady increase throughout the year, ending at \$817,860.05 on December 31, 2015.

9. Each pizza category contributed as follows to the total revenue:

Classic: 26.91%

Supreme: 25.46%

Chicken: 23.96%

Veggie: 23.68%





Insights

10. Top Pizzas by Revenue (Chicken, Classic, Supreme, Veggie):

Chicken Category:

1. The Thai Chicken Pizza - \$43,434.25
2. The Barbecue Chicken Pizza - \$42,768.00
3. The California Chicken Pizza - \$41,409.50

Classic Category:

1. The Classic Deluxe Pizza - \$38,180.50
2. The Hawaiian Pizza - \$32,273.25
3. The Pepperoni Pizza - \$30,161.75

Supreme Category:

1. The Spicy Italian Pizza - \$34,831.25
2. The Italian Supreme Pizza - \$33,476.75
3. The Sicilian Pizza - \$30,940.50

Veggie Category:

1. The Four Cheese Pizza - \$32,265.70
2. The Mexicana Pizza - \$26,780.75
3. The Five Cheese Pizza - \$26,066.50



Recommendations

1. Focus marketing efforts on The Greek Pizza (highest priced pizza) and the top-performing pizzas like The Classic Deluxe Pizza and The Barbecue Chicken Pizza to drive sales based on revenue and popularity.
2. Review the pricing strategy for less popular pizzas to make them more appealing to customers.
3. Enhance promotions and staffing during peak hours (12 PM, 1 PM, 6 PM, and 5 PM) to maximize sales and improve customer service.
4. Given the popularity of Large (Size L) pizzas, consider expanding the variety in this size category to cater to customer preferences.
5. Develop targeted marketing campaigns for each pizza category to boost their visibility and sales, particularly for categories with lower revenue contributions.
6. Introduce seasonal promotions or special offers for underperforming pizzas to stimulate interest and increase sales.
7. Collect and analyze customer feedback to understand preferences better and make data-driven decisions for menu adjustments and new product introductions.

