



Pizza Sales Analysis Using MySQL

Presented By Sanskar Choudhari

ABOUT PROJECT

This project focuses on analyzing a pizza sales dataset using **MySQL** to extract meaningful business insights and support data-driven decision-making. The goal was to explore sales patterns, product performance, and customer behavior through SQL queries ranging from basic aggregations to advanced analytics.

The analysis was conducted in MySQL, and all results, including SQL query screenshots and visual summaries, have been compiled into this presentation to demonstrate technical proficiency and real-world data application.

- Retrieve the total number of orders placed.

```
-- Retrieve the total number of orders placed.  
  
SELECT  
    COUNT(order_id) AS total_orders  
FROM  
    orders;
```

Query

Result Grid	
	total_orders
▶	21350

Solution

- Calculate the total revenue generated from pizza sales.

```
-- Calculate the total revenue generated from pizza sales.
```

```
SELECT
```

```
    ROUND(SUM(order_details.quantity * pizzas.price),  
           2) AS total_sales
```

```
FROM
```

```
    order_details
```

```
    JOIN
```

```
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

Result Grid	
	total_sales
▶	817860.05

➤ Identify the highest-priced pizza.

```
-- 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;
```

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

- Identify the most common pizza size ordered.

```
-- Identify the most common pizza size ordered.
```

```
SELECT
```

```
    pizzas.size,
```

```
    COUNT(order_details.order_details_id) AS order_count
```

```
FROM
```

```
    pizzas
```

```
    JOIN
```

```
    order_details ON pizzas.pizza_id = order_details.pizza_id
```

```
GROUP BY pizzas.size
```

```
ORDER BY order_count DESC;
```

Result Grid			Filter
	size	order_count	
▶	L	18526	
	M	15385	
	S	14137	
	XL	544	
	XXL	28	

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

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

SELECT
    pizza_types.name, 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.name
ORDER BY quantity DESC
LIMIT 5;
```

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	

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

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

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

- **Determine the distribution of orders by hour of the day.**

```
-- Determine the distribution of orders by hour of the day.
```

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

Result Grid			Filter Rows:
	hour(order_time)	count(order_id)	
▶	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.

```
-- Join relevant tables to find the category-wise distribution of pizzas.  
  
SELECT  
    category, COUNT(name)  
FROM  
    pizza_types  
GROUP BY category;
```

Result Grid			Filter Rows:
	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.**

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

SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
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;
```

Result Grid		Filter Rows:
	avg_pizza_ordered_per_day	
▶	138	

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

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

SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),
          0) AS revenue
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 revenue DESC
LIMIT 3;
```

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 type to total revenue.

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

SELECT
    pizza_types.category,
    ROUND((SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
    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 order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

Result Grid			Filter
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

➤ Analyze the cumulative revenue generated over time.

```
-- Analyze the cumulative revenue generated over time.

SELECT
    order_date,
    ROUND(SUM(revenue) OVER (ORDER BY order_date), 2) AS cum_revenue
FROM (
    SELECT
        o.order_date,
        ROUND(SUM(od.quantity * p.price), 2) AS revenue
    FROM
        order_details od
    JOIN
        pizzas p ON od.pizza_id = p.pizza_id
    JOIN
        orders o ON o.order_id = od.order_id
    GROUP BY
        o.order_date
) AS sales;
```

Result Grid			Filter Rows:
	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	

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

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

SELECT
    name,
    ROUND(revenue, 2) AS revenue
FROM (
    SELECT
        category,
        name,
        revenue,
        RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS rn
    FROM (
        SELECT
            pt.category,
            pt.name,
            SUM(od.quantity * p.price) AS revenue
        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,
            pt.name
    ) AS a
    ) AS b
WHERE
    rn <= 3;
```

Result Grid			Filter Rows:
	name	revenue	
▶	The Thai Chicken Pizza	43434.25	
	The Barbecue Chicken Pizza	42768	
	The California Chicken Pizza	41409.5	
	The Classic Deluxe Pizza	38180.5	
	The Hawaiian Pizza	32273.25	
	The Pepperoni Pizza	30161.75	
	The Spicy Italian Pizza	34831.25	
	The Italian Supreme Pizza	33476.75	
	The Sicilian Pizza	30940.5	
	The Four Cheese Pizza	32265.7	
	The Mexicana Pizza	26780.75	
	The Five Cheese Pizza	26066.5	

Strategies for growth

KEY STRATEGIES :

- Identify top-performing products to boost marketing focus.
- Determine customer preferences by time and size.
- Leverage high-revenue pizza types for bundled offers.
- Align kitchen operations with peak ordering hours.

Marketing strategies

Insights:

- **Category-Wise Quantity Analysis** helps decide what type of pizzas to scale up.
- **Hourly Distribution of Orders** guides expansion of delivery hours or promotions.
- **Category Distribution** informs inventory and staffing.

Opportunities:

- Promote popular categories during high order hours.
- Expand offerings in underperforming categories.

The background of the slide is a light blue gradient with several large, soft, 3D-style bubbles in shades of blue and purple. These bubbles are of various sizes and are scattered across the frame, with some appearing more prominent than others. The overall aesthetic is clean, modern, and visually appealing.

Innovative solutions

Solutions :

- Focus on high-revenue items for premium promotions.
- Reconsider stock for low-contribution pizzas.

Sales Targets

Key Metric :

Average Pizzas Ordered
Per Day : **138**

Use Case :

Set weekly or monthly pizza sales targets based on the average daily sales to monitor store performance.

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

SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
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;
```

Result Grid		Filter Rows:
	avg_pizza_ordered_per_day	
▶	138	

Future initiatives

Cumulative Revenue Over Time

Top 3 Revenue Pizzas by Category :

The Thai Chicken Pizza
The Barbecue Chicken Pizza
The California Chicken Pizza

Future Plans:

Dynamic pricing based on trends.
Automated dashboards for daily sales monitoring.
Geo-based performance tracking with branch comparison.

Result Grid			Filter Rows:
	name	revenue	
▶	The Thai Chicken Pizza	43434.25	
	The Barbecue Chicken Pizza	42768	
	The California Chicken Pizza	41409.5	
	The Classic Deluxe Pizza	38180.5	
	The Hawaiian Pizza	32273.25	
	The Pepperoni Pizza	30161.75	
	The Spicy Italian Pizza	34831.25	
	The Italian Supreme Pizza	33476.75	
	The Sicilian Pizza	30940.5	
	The Four Cheese Pizza	32265.7	
	The Mexicana Pizza	26780.75	
	The Five Cheese Pizza	26066.5	

Thank you

➤ Connect With Me :

- **Github :** <https://github.com/sanskarchoudhari-1>
- **Email :** sanskarchoudhari3@gmail.com
- **Linkedin :** www.linkedin.com/in/sanskar-choudhari-720729207