

Pizza Parlor :

SQL Data Analysis

Project

By- Sakshi Mogal





Contents

1. Project Title
2. Table of Contents
3. Query Utilization
4. Data Overview and Contextual Analysis
5. Analytical Framework for Questions
 - Foundational Queries (Basic Level)
 - Intermediate Analytical Queries
 - Advanced Analytical Queries
6. Foundational Insights: Queries and Outputs
7. Intermediate-Level Insights: Queries and Outputs
8. Advanced-Level Insights: Queries and Outputs
9. Key Observations and Insights
10. Conclusion and Future Recommendations

How I Utilized SQL Queries

1. Data Extraction and Retrieval

- Extracted relevant data from large datasets using commands like SELECT, WHERE, and JOIN.
- Combined data from multiple tables for a comprehensive analysis.

2. Data Aggregation and Summarization

- Applied aggregate functions such as SUM, COUNT, AVG.
- Used GROUP BY to summarize data and derive key insights like trends and performance metrics.

3. Ranking and Classification

- Implemented ranking functions RANK() to classify data.
- Identified top performers, trends, and segmented records effectively.

4. Conditional Logic for Analysis

- Used CASE statements for dynamic classification and conditional calculations.
- Enabled enhanced insights, such as creating categories or generating custom outputs.

5. Performance Optimization

- Optimized query performance through indexing, subqueries.,
- Ensured faster execution and accurate results, even on complex datasets.



Data Overview

1

pizzas
pizza_id
pizza_type_id
Σ price
size

2

pizza_types
category
ingredients
name
pizza_type_id

3

orders
date
order_id
time

4

order_details
Σ order_details_id
order_id
pizza_id
Σ quantity



Questions

Foundational Queries (Basic Level)

- ✓ 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.

Intermediate Analytical Queries

- ✓ 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.

Advanced Analytical Queries

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



Foundational Queries

Retrieve the total number of orders placed.

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

Total_orders
21350



Calculate the total revenue generated from pizza sales.

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

Result Grid	
	Total_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.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
LIMIT 3;
```

	size	order_count
▶	L	18526
	M	15385
	S	14137



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;
```

	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



Intermediate Queries

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;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



Determine the distribution of orders by hour of the day.

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

	hours	order_count
▶	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.

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

	distribution_of_pizza	category
▶	6	Chicken
	8	Classic
	9	Supreme
	9	Veggie



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

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

average_pizzas_order_per_day
138





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

```
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;
```



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



Advanced Queries

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 order_details.pizza_id = pizzas.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;
```

category	Revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68



Calculate the percentage contribution of top 5 pizza to total revenue.

```
SELECT
    pizza_types.name,
    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 order_details.pizza_id = pizzas.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.name
ORDER BY Revenue DESC
LIMIT 5;
```



	name	Revenue
▶	The Thai Chicken Pizza	5.31
	The Barbecue Chicken Pizza	5.23
	The California Chicken Pizza	5.06
	The Classic Deluxe Pizza	4.67
	The Spicy Italian Pizza	4.26





Analyze the cumulative revenue generated over time.

```
select order_date,  
       sum(revenue) over(order by order_date) as cum_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	cum_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	23990.35000000002
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.30000000003
	2015-01-14	32358.70000000004
	2015-01-15	34343.50000000001
	2015-01-16	36937.65000000001
	2015-01-17	39001.75000000001
	2015-01-18	40978.60000000006
	2015-01-19	43365.75000000001
	2015-01-20	45763.65000000001
	2015-01-21	47804.20000000001
	2015-01-22	50300.90000000001



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

```
select category, name, revenue, pizza_rank from
( select category ,name,revenue,
rank() over(partition by category order by revenue desc) as pizza_rank from
( select pizza_types.category,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.category,pizza_types.name) as a )as b
where pizza_rank <=3;
```

	category	name	revenue	pizza_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 Observations and Insights



• **Customer Preferences:**

- Customers love The Classic deluxe pizza & The barbecue Chicken pizza
- Also they Prefer are L and M sizes.

• **Revenue Insights:**

- High-revenue pizzas should be prioritized in marketing.
- The Thai Chicken pizza, the Barbecue chicken pizza and the California Chicken pizza generates the highest revenue

• **Operational Efficiency:**

- Hourly order distribution helps optimize staffing and delivery schedules.
- Afternoon 12 to 2 pm & evening 5 to 9 pm are the most ordered hours

• **Customer Loyalty:**

- Repeat orders and popular items can help retain customers and boost sales.
- Most popular pizza categories are Classic and Supreme pizzas

Thank you!

Do you have any questions?
sakshimogal1607@gmail.com

