

PIZZA SALES ANALYSIS USING SQL



HELLO !

Hello, my name is Arpit Awasthi, and in this project, I utilized SQL queries ranging from basic to advanced levels to solve questions related to pizza sales. The aim was to uncover meaningful insights and trends that would aid in making informed business decisions and optimizing future sales strategies.

Arpit Awasthi





QUESTIONS RELATED TO PIZZA SALES

Basic:

- 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.

Intermediate:

- 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.

Advanced:

- 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.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

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

OUTPUT

Result Grid	
	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;`

OUTPUT

Result Grid	
	<code>total_revenue</code>
▶	<code>817860.05</code>

IDENTIFY THE HIGHEST-PRICED PIZZA

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

OUTPUT

Result Grid | Filter Rows:

	Highest_rated_pizza	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 1;
```

OUTPUT

Result Grid | Filter

	size	order_count
▶	L	18526

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;
```

OUTPUT

	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

```
SELECT  
    pizza_types.category AS 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 Category  
ORDER BY quantity DESC;
```

OUTPUT

Result Grid | Filter

	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 hour,  
    COUNT(order_id) AS order_distribution  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

OUTPUT

	hour	order_distribution
▶	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

FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```
SELECT  
    category, COUNT(pizza_type_id) AS pizza_count  
FROM  
    pizza_types  
GROUP BY category;
```

OUTPUT

Result Grid | Filter Rows

	category	pizza_count
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

SELECT

```
    ROUND(AVG(quantity), 0) AS average_pizzas_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 order_date) AS a;
```

OUTPUT

Result Grid | Filter Rows:

	average_pizzas_ordered_per_day
▶	138

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),
        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 3;
```

OUTPUT

Result Grid | Filter Rows:

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

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_revenue
    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;
```

OUTPUT

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
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 orders join order_details on orders.order_id = order_details.order_id join pizzas on pizzas.pizza_id = order_details.pizza_id group by orders.order_date) as total_revenue;
```

OUTPUT

Result Grid		
	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
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.300000000003
	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
	2015-01-23	52724.60000000006

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```
select name, revenue from
(select category,name,revenue, rank() over(partition by category order by revenue desc) as rn from
(select pizza_types.category,pizza_types.name,ROUND(SUM(order_details.quantity * pizzas.price),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,pizza_types.name
ORDER BY revenue DESC) as a) as b
where rn <=3;
```

OUTPUT

	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

KEY TAKEAWAYS:

1) XXL Pizza Size Underperformance:

Out of 21,350 orders, XXL pizzas were ordered only 28 times. It's recommended to discontinue this size and focus on more popular sizes.

2) Classic Pizza Popularity:

Among the four pizza categories, customers prefer the "Classic" category the most, making it a key area to focus on.

3) Order Timing Trend:

There are very few orders between 9 AM and 11 AM, with peak sales starting from 11 PM, likely due to people being busy during the day.

4) Chicken Pizza Category:

The chicken category has only six pizza types, leading to fewer sales. Expanding the variety of chicken pizzas, like other categories (supreme, veggies, classic), could boost sales.

5) Average Orders:

With an average of 138 pizzas sold per order, further advertising, discounts, or coupons could help increase this number and attract more customers.

THANK YOU!

