

PROJECT BY-AYUSHYA KUMAR

ABOUT THIS PROJECT





This project analyzes pizza sales data using SQL to derive meaningful business insights. It involves cleaning and structuring raw sales data, performing complex queries to extract key metrics, and generating reports to assist in decision-making. Key analyses include:

- Total revenue and sales trends over time
- Best-selling and least-selling pizza categories
- Order frequency and peak sales hours
- Customer purchase behavior and average order value
- Performance comparison of different pizza sizes and toppings

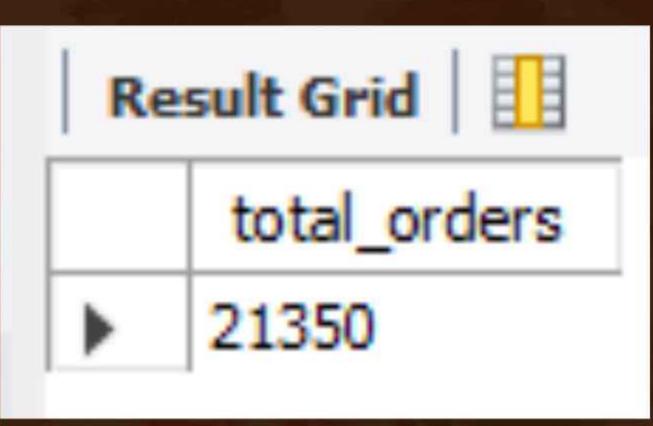
By leveraging SQL queries such as aggregation, joins, subqueries, and window functions, this project provides data-driven insights to optimize sales strategies and improve business efficiency.

QUESTIONS

- 1) Retrieve the total numbers of order 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 categor-wise distributtion 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 oln revenue for each pizza category

RETRIEVE THE TOTAL NUMBERS OF ORDER PLACED





CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```
SELECT

ROUND(SUM(order_details.quantity * pizzas.price),

2) AS total_sales

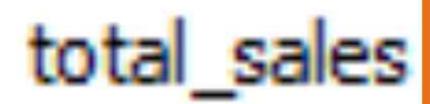
FROM

order_details

JOIN

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





817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA

name

price

The Greek Pizza

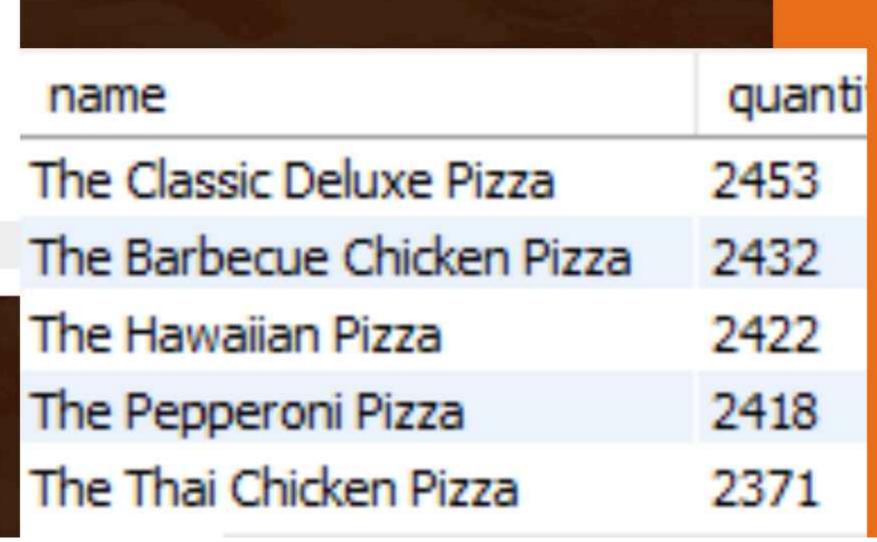
35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

size	order_count
L	18526
М	15385
S	14137
XL	544
XXL	28

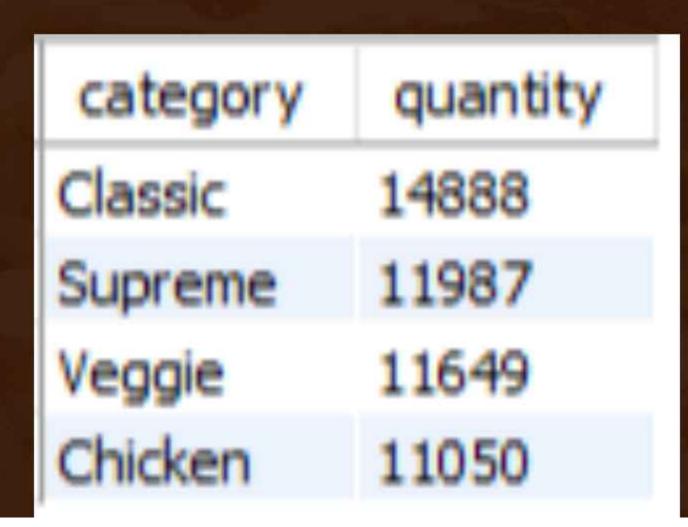
LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```
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;
```



JOIN THE NECESARY 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;
```



DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM

orders

GROUP BY HOUR(order_time)

ORDER BY order_count DESC;
```

hour	order_count
21	1198
22	663
23	28
10	8
9	1

JOIN THE RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```
SELECT

category, COUNT(name)

FROM

pizza_types

GROUP BY category;
```

category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERES 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;
```

avg_pizza_ordered_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 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;

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 sales
                FROM
                    order_details
                        JOIN
                    pizzas ON pizzas.pizza id = order details.pizza id) * 100
            2) 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.category
ORDER BY revenue DESC;
```

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 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	23990.3500000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006



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,
        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 rn<=3;
```

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	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5



THANK YOU FOR ATTENTION

ayushcom.kumar @gmail.com www.linkedin.com/in /ayushya-kumar06