



SQL PROJECT

PIZZA SALES

20 OCTOBER, 2024



INTRODUCTION

This project involves analyzing a pizza sales dataset using SQL. The dataset consists of four CSV files: `order_details`, `orders`, `pizzas`, and `pizza_types`. Through SQL queries ranging from beginner to advanced levels, various business insights were extracted, such as identifying popular pizza types, tracking sales trends. By integrating these tables, the project covers essential operations like joining tables, aggregating data, and answering complex business questions related to pizza orders and sales performance.

-- Retrieve the total number of orders placed.

```
select count(order_id) as total_order from orders;
```

total_order
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;

total_revenue
817860.05

--- Identify the highest-priced pizza.

- **SELECT**
 pizza_types.pizza_name, pizzas.price
FROM
 pizza_types
 JOIN
 pizzas **ON** pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY pizzas.price **DESC**
LIMIT 1;

pizza_name	price
The Greek Pizza	35.95

-- Identify the most common pizza size ordered.

- **SELECT**

```
pizzas.size, COUNT(order_details.quantity) AS order_count
```

```
FROM
```

```
pizzas
```

```
JOIN
```

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

```
GROUP BY pizzas.size
```

```
ORDER BY COUNT(order_details.quantity) DESC;
```

	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.

- **SELECT**

```
    pizza_types.pizza_name,  
    COUNT(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.pizza_name  
ORDER BY COUNT(order_details.quantity) DESC  
LIMIT 5;
```

pizza_name	quantity
The Classic Deluxe Pizza	2416
The Hawaiian Pizza	2370
The Pepperoni Pizza	2369
The Thai Chicken Pizza	2315
The California Chicken Pizza	2302

-- 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;
```

	category	quantity	
	Classic	14888	
	Veggie	11649	
	Supreme	11987	
	Chicken	8618	

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

- **SELECT**

```
HOUR(order_time) as hour, COUNT(order_id) AS count
```

```
FROM
```

```
orders
```

```
GROUP BY HOUR(orders.order_time);
```

hour	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
    category, COUNT(pizza_name) AS count
FROM
    pizza_types
GROUP BY category;
```

	category	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_quantity_ordered
```

```
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;
```

average_quantity_ordered
138

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

- **SELECT**

```
    pizza_types.pizza_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.pizza_name  
ORDER BY revenue DESC  
LIMIT 3;
```

	pizza_name	revenue
	The Thai Chicken Pizza	43434.25
	The California Chicken Pizza	41409.5
	The Classic Deluxe Pizza	38180.5

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

```
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
        SUM(order_details.quantity * pizzas.price) AS total_sales
    FROM
        order_details
        JOIN
            pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,
    0) 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	27
Supreme	25
Veggie	24
Chicken	19

-- Analyze the cumulative revenue generated over time.

- ```
select order_date, sum(reveneu) over (order by order_date) as cum_reveneu
from
(select orders.order_date,
round(sum(order_details.quantity * pizzas.price),2) as reveneu
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 sales;
```

And the order\_date, cumulative\_sum continue till...

|  | order_date | cum_reveneu |
|--|------------|-------------|
|  | 2015-12-26 | 809196.8    |
|  | 2015-12-27 | 810615.8    |
|  | 2015-12-28 | 812253      |
|  | 2015-12-29 | 813606.25   |
|  | 2015-12-30 | 814944.05   |
|  | 2015-12-31 | 817860.05   |

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

- ```
select category, pizza_name, revenue  
from  
(select category, pizza_name, revenue,  
rank() over (partition by category order by revenue desc) as ranking  
from  
(SELECT  
    pizza_types.category,  
    pizza_types.pizza_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.pizza_name) as overall  
where ranking <= 3;
```

category	pizza_name	revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The California Chicken Pizza	41409.5
Chicken	The Southwest Chicken Pizza	34705.75
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



THANK YOU

20 october, 2024