



PIZZA SALES ANALYSIS USING SQL

-By Puspita Chowdhury



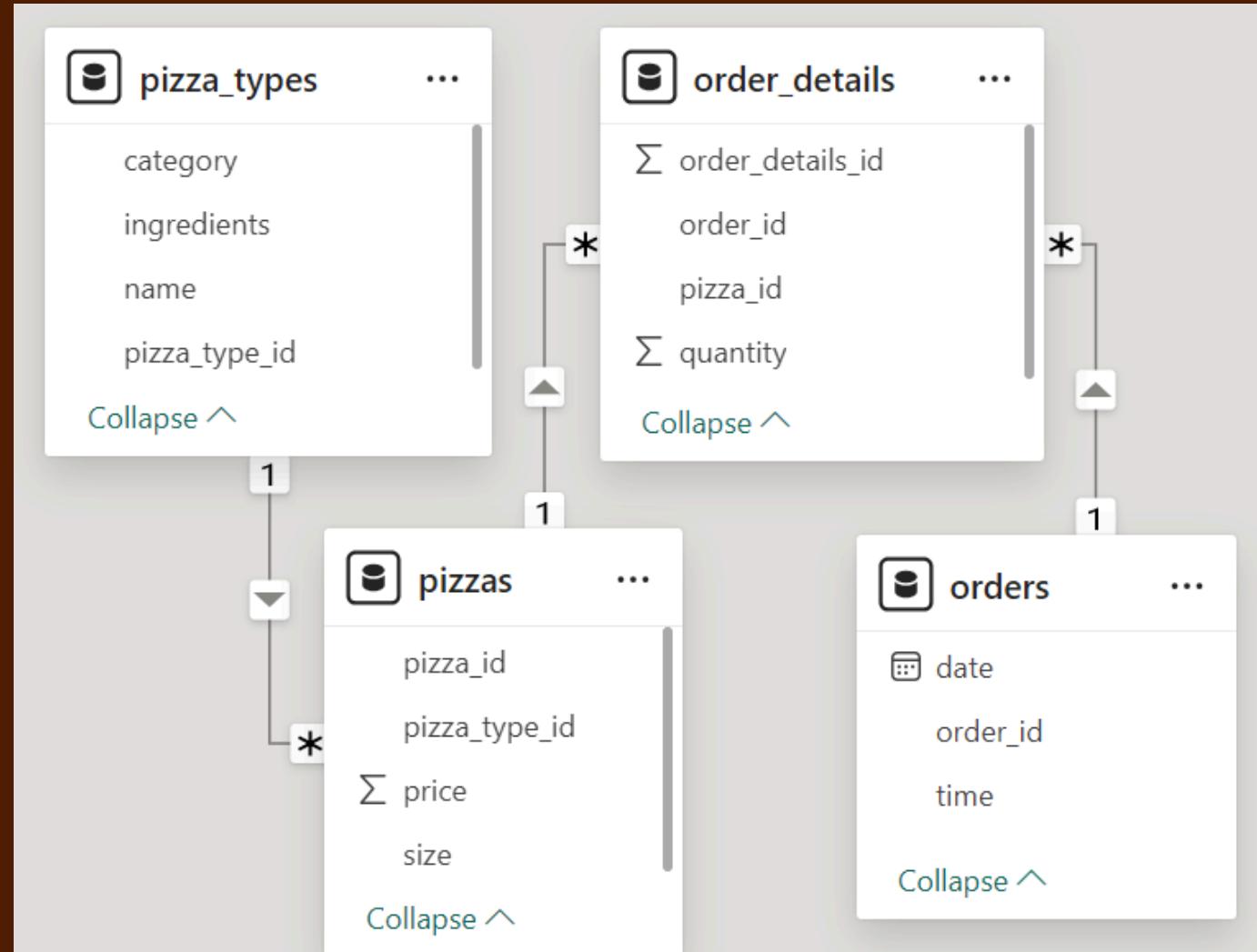
Hello !!!

This is Puspita Chowdhury. In this project, I have utilised basic and intermediate level SQL queries to solve questions that were related to Pizza Sales.

Dataset Description

- `orders_details.csv` - Contains transaction details as order detail id, order id, pizza id, quantity.
- `orders.csv` - Contains order information like order id, order date and order time.
- `pizza_types.csv` - Contains details about pizza name, category, ingredients.
- `pizzas.csv` - Contains information about pizza type and size

Schema Diagram





Questions to be answered

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.
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.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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

Result Grid	
	total_orders
	21350

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 order_details.pizza_id = pizzas.pizza_id;
```

Result Grid	
	total_sales
	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 price DESC

LIMIT 1;

Result Grid |   Filter

	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
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY size
ORDER BY order_count 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.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

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 category  
ORDER BY quantity DESC;
```

Result Grid

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

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

	HOUR(order_time)	COUNT(order_id)
	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(name)

FROM

pizza_types

GROUP BY category;

Result Grid | Filter Row

	category	COUNT(name)
1	Chicken	6
2	Classic	8
3	Supreme	9
4	Veggie	9

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

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

The screenshot shows a database query results interface. On the left, there is a code editor with the SQL query. To the right, there are two tabs: "Result Grid" and "SQL". The "Result Grid" tab is selected, displaying a single row of results. The column header is "avg_pizza_orders" and the value is "138".

avg_pizza_orders
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;
```

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbeque Chicken Pizza	42768
The California Chicken Pizza	41409.5



Thanks !!!