Pizza Sales Report

JUNE 2024

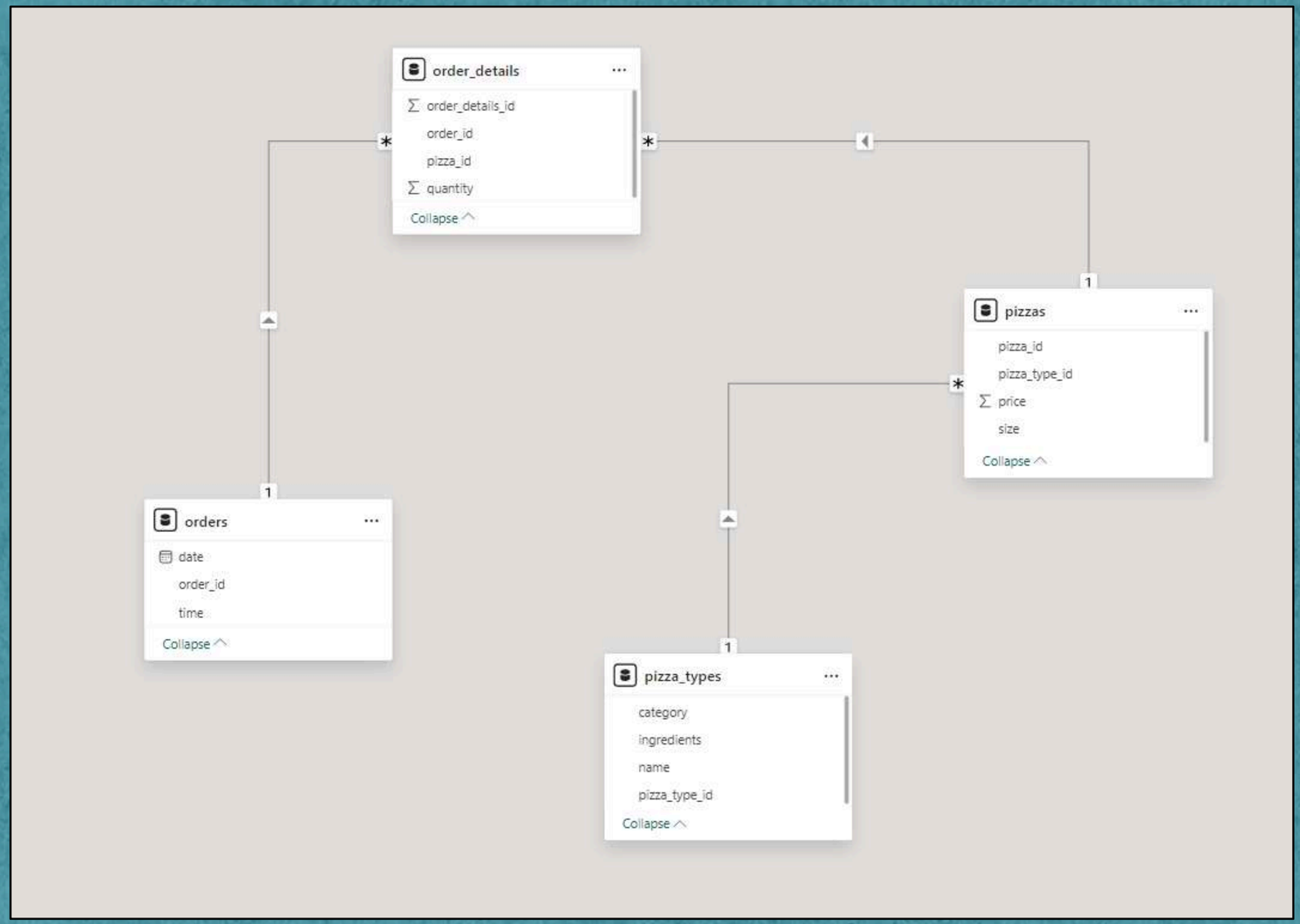
Introduction

Hello, I am Anchal kumari, a third year student pursuing BTECH in Production and Industrial Engineering.
In this project I have utilized Sql queries to solve the questions that were related to pizza sales.

Objective :

In the fast-paced world of the food and beverage industry, managing sales data efficiently is crucial for making informed business decisions. This project focuses on creating an SQL database for managing and analyzing pizza sales, enabling the business to keep track of orders, customer preferences, and sales performance.
It organize the data, track the sales, manages the inventory and analyse the performance.

Model View :


order_details
order_details_id
order_id
pizza_id
quantity
Collapse
pizzas
pizza_id
pizza_type_id
price
size
Collapse
orders
date
order_id
time
Collapse
pizza_types
category
ingredients
name
pizza_type_id
Collapse
1
1
1

Retrieve the total number of orders placed.

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

Result Grid

	total_orders
▶	21350

Calculate the total revenue generated from pizza sales.

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

Result Grid

total_revenue
513563.2

Identify the highest-priced pizza.

```sql
SELECT
    pizza_types.name, pizza_types.pizza_type_id, 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 Rows:

name	pizza_type_id	price
The Greek Pizza	the_greek	35.95

Identify the most common pizza size ordered.

```sql
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;
```

Result Grid | Filter Rows:

size	order_count
L	11651
M	9674
S	8800
XL	357
XXL	18

List the top 5 most ordered pizza types along with their quantity.

```sql
SELECT
    pizza_types.name, SUM(order_details.quantity)
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY name
ORDER BY SUM(order_details_id) DESC
LIMIT 5;
```

Result Grid | Filter Rows: | Export:

name	sum(order_details.quantity)
The Classic Deluxe Pizza	1504
The Barbecue Chicken Pizza	1560
The California Chicken Pizza	1489
The Hawaiian Pizza	1493
The Pepperoni Pizza	1501

Join the necessary tables to find the total quantity of each pizza category ordered.

```sql
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY category
ORDER BY quantity DESC;
```

Result Grid | Filter Rows:

category	quantity
Classic	9270
Supreme	7540
Veggie	7380
Chicken	6906

Determine the distribution of orders by hour of the day.

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

Result Grid | Filter Rows:

HOUR(time)	COUNT(order_id)
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

Join relevant tables to find the category-wise distribution of pizzas.

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

Result Grid | Filter Rows:

category	count(pizza_type_id)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

```sql
SELECT
    pizza_types.name,
    SUM(pizzas.price * order_details.quantity) AS revenue
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

Result Grid | Filter Rows:

name	revenue
The Barbecue Chicken Pizza	27418
The Thai Chicken Pizza	26698.25
The California Chicken Pizza	26076.75

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


```sql
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    SUM(order_details.quantity * pizzas.price) AS sale
                FROM
                    order_details
                        JOIN
                    pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,
            2) AS percent
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 percent;
```

Result Grid | Filter Rows:

category	percent
Chicken	23.84
Veggie	23.91
Supreme	25.54
Classic	26.71

Thank You!