Pizza Sales Analysis

An SQL Query-Based Project

K.Shyam Sundar

Introduction

This is shyam sundar.

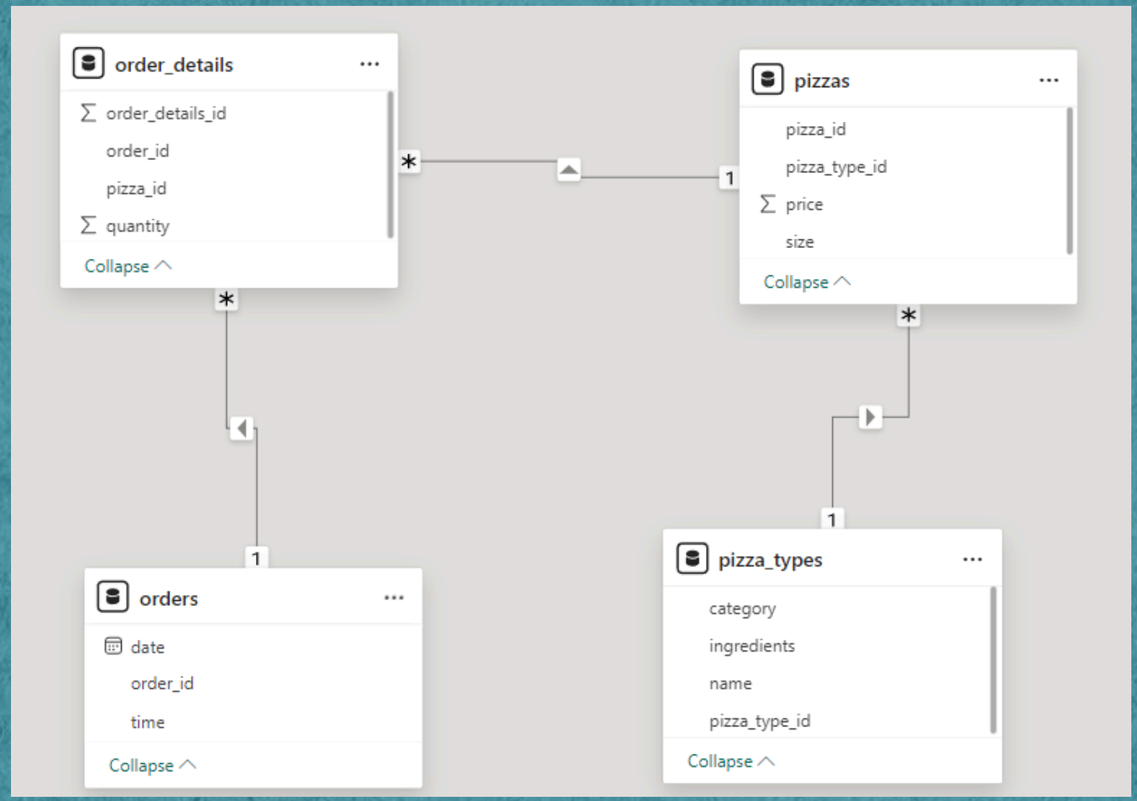
In this project I have utilized SQL queries to solve questions related to pizza sales.

Database Overview

The pizza sales database is designed to capture and analyze various aspects of pizza sales, including order details, customer orders, types of pizzas, and their attributes. This database is structured to provide comprehensive insights into sales performance, customer preferences, and revenue distribution.

Key Tables Used for Analysis:

1. Order_details
2. Orders
3. Pizza_types
4. Pizzas



Retrieve the Total Number of Orders Placed

```sql
Select count(order_id) as total_orders from orders;
```

total_orders
21350

Calculate the Total Revenue Generated from Pizza Sales

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

Total_sales
817860.05

Identify the Highest-Priced Pizza

```sql
Select pizza_types.name, pizzas.price
from pizza_types JOIN pizzas
ON pizza_types.pizza_type_id=pizzas.pizza_type_id
Where price=(select max(price) from pizzas);
```

	name	price
▶	The Greek Pizza	35.95

Identify the Most Common Pizza Size Ordered

```sql
Select COUNT(orders_details.quantity) AS order_count ,pizzas.size
from orders_details JOIN pizzas
ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY pizzas.size order BY order_count DESC;
```

order_count	size
18526	L

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

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

name	Quantities
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.

```sql
select SUM(orders_details.quantity) as Quantity, pizza_types.category
FROM pizza_types
JOIN pizzas ON pizza_types.pizza_type_id=pizzas.pizza_type_id
JOIN orders_details ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY category
order by Quantity DESC;
```

Quantity	category
14888	Classic
11987	Supreme
11649	Veggie
11050	Chicken

Determine the distribution of orders by hour of the day.

```
Select count(order_id) as Order_count, hour(order_time) as hour from orders
GROUP BY hour(order_time);
```

	Order_count	hour
▶	1231	11
	2520	12
	2455	13
	1472	14
	1468	15
	1920	16
	2336	17
	2399	18
	2009	19
	1642	20
	1198	21
	663	22
	28	23
	8	10
	1	9

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

```
select category, COUNT(name) AS Distribution_of_Pizzas from  pizza_types
group by category;
```

category	Distribution_of_Pizzas
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 ROUND(avg(quantity),0) as Avg_pizzas_ordered_per_day
from(
select orders.order_date, SUM(orders_details.quantity) as quantity
from orders_details
JOIN orders
ON orders.order_id=orders_details.order_id
GROUP BY order_date) AS order_quantity;
```

	Avg_pizzas_ordered_per_day
▶	138

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

```sql
select pizza_types.name, SUM(orders_details.quantity*pizzas.price) as revenue
from pizza_types
JOIN pizzas
ON  pizza_types.pizza_type_id = pizzas.pizza_type_id
JOIN orders_details
ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY 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.

```sql
select pizza_types.category,
round(sum(orders_details.quantity*pizzas.price)/                    ## total reveune = Quantit* price
(select ROUND(SUM(orders_details.quantity*pizzas.price),2) as total_sales
from orders_details JOIN pizzas ON orders_details.pizza_id=pizzas.pizza_id)*100,2) as Cal_revenue
from pizzas
JOIN pizza_types
ON pizza_types.pizza_type_id = pizzas.pizza_type_id
JOIN orders_details
ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY category ORDER BY Cal_revenue DESC;
```

	category	Cal_revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

Analyze the cumulative revenue generated over time.

```sql
select order_date, SUM(Revenue) over (order by order_date) as CUM_Revenue
from
(Select orders.order_date, SUM(orders_details.quantity*pizzas.price) as revenue
from orders_details
JOIN pizzas
ON orders_details.pizza_id = pizzas.pizza_id
JOIN orders
ON orders.order_id = orders_details.order_id
GROUP BY order_date) as Sales;
```

order_date	CUM_Revenue
2015-01-01 00:00:00	2713.8500000000004
2015-01-02 00:00:00	5445.75
2015-01-03 00:00:00	8108.15
2015-01-04 00:00:00	9863.6
2015-01-05 00:00:00	11929.55
2015-01-06 00:00:00	14358.5
2015-01-07 00:00:00	16560.7
2015-01-08 00:00:00	19399.05
2015-01-09 00:00:00	21526.4
2015-01-10 00:00:00	23990.350000000002
2015-01-11 00:00:00	25862.65
2015-01-12 00:00:00	27781.7
2015-01-13 00:00:00	29831.300000000003
2015-01-14 00:00:00	32358.700000000004
2015-01-15 00:00:00	34343.50000000001
2015-01-16 00:00:00	36937.65000000001
2015-01-17 00:00:00	39001.75000000001
2015-01-18 00:00:00	40978.600000000006
2015-01-19 00:00:00	43365.75000000001

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

```sql
select name, revenue, ranke
from
(select category,name, revenue,
rank() over (partition by category order by revenue desc) as ranke
from
(select pizza_types.name,pizza_types.category,SUM(orders_details.quantity*pizzas.price) as revenue
from pizza_types
JOIN pizzas
ON pizza_types.pizza_type_id=pizzas.pizza_type_id
JOIN orders_details
ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY pizza_types.name,pizza_types.category) as table_1) as table_2
where ranke<=3;
```

name	revenue	ranke
The Thai Chicken Pizza	43434.25	1
The Barbecue Chicken Pizza	42768	2
The California Chicken Pizza	41409.5	3
The Classic Deluxe Pizza	38180.5	1
The Hawaiian Pizza	32273.25	2
The Pepperoni Pizza	30161.75	3
The Spicy Italian Pizza	34831.25	1
The Italian Supreme Pizza	33476.75	2
The Sicilian Pizza	30940.5	3
The Four Cheese Pizza	32265.70000000065	1
The Mexicana Pizza	26780.75	2
The Five Cheese Pizza	26066.5	3

Conclusion

This project utilized SQL queries to analyze pizza sales data, revealing key insights into customer preferences and sales performance.

- **Comprehensive Sales Overview: Determined the total number of orders and overall revenue, providing a clear picture of the business's financial health.**
- **Customer Preferences: Identified the highest-priced pizza and the most common pizza size, highlighting key customer preferences.**
- **Top-Selling Products: Revealed the top five most ordered pizza types, aiding targeted marketing and inventory management.**
- **Trend Analysis: Examined order distributions by hour and date, identifying peak sales times and trends in pizza consumption.**
- **Revenue Insights: Calculated the percentage contribution of each pizza type to total revenue and identified key revenue drivers within each category, supporting strategic decision-making.**

Thank you!