

SQL PROJECT

PIZZA SALES



Pizza Sales Analysis Project Presentation

Objective:

The primary objective of this project is to analyze sales data related to pizza orders and provide insights to enhance decision-making processes for the pizza business. Specifically, the objectives are:

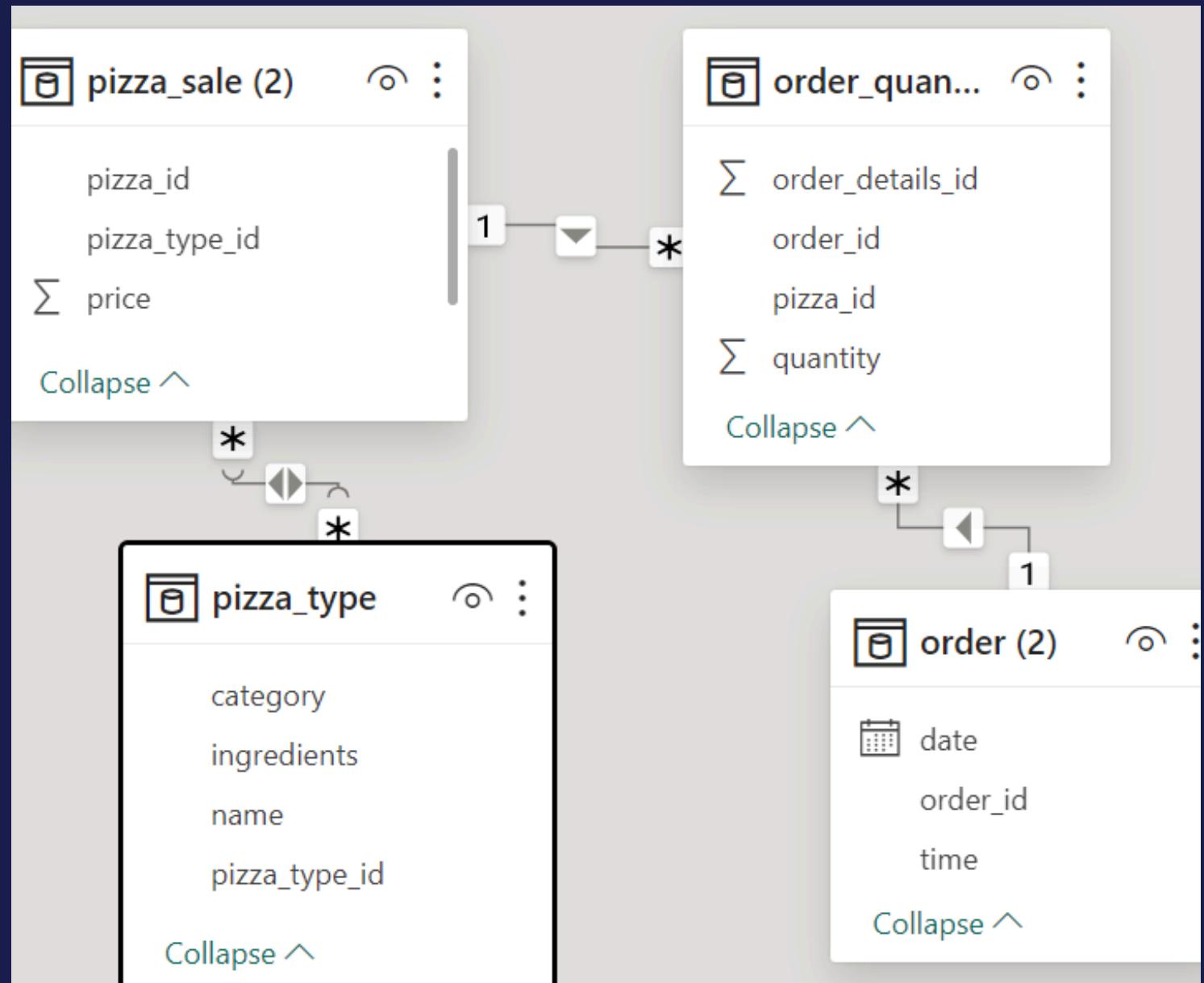
1. Understand the overall sales performance by retrieving key metrics such as total orders placed and total revenue generated.
2. Identify patterns and trends in pizza orders, including popular pizza sizes, pizza types, and distribution of orders over time.
3. Analyze revenue contribution and performance of different pizza types and categories.
4. Provide recommendations for improving sales and optimizing product offerings based on the analysis results.

Outcome:

Through this project, we aim to achieve the following outcomes:

1. Gain a comprehensive understanding of the pizza sales data and its characteristics.
2. Identify actionable insights that can drive business decisions and strategies.
3. Develop visualizations and presentations to effectively communicate the analysis results.
4. Provide recommendations for future actions to improve sales performance and customer satisfaction.

MODEL VIEW



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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

Result Grid	
	Total_sales
▶	817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA.

SELECT

```
    pizza_type_id, MAX(price) AS piy  
FROM  
    pizzas  
GROUP BY pizza_type_id  
ORDER BY piy DESC  
LIMIT 1;
```

Result Grid



Filter

	pizza_type_id	piy
▶	the_greek	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT  
    size, COUNT(size) AS maximum_size_ordered_pizza  
FROM  
    pizzas  
GROUP BY size  
ORDER BY maximum_size_ordered_pizza DESC  
LIMIT 1;
```

Result Grid | Filter Rows:

	size	maximum_size_ordered_pizza
▶	S	32

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

SELECT

```
    pizza_type_id, SUM(quantity) AS ordered_quantity  
FROM  
    pizzas p  
        JOIN  
    order_details o ON o.pizza_id = p.pizza_id  
        GROUP BY pizza_type_id  
ORDER BY ordered_quantity DESC  
LIMIT 5;
```

	pizza_type_id	ordered_quantity
▶	classic_dlx	2453
	bbq_ckn	2432
	hawaiian	2422
	pepperoni	2418
	thai_ckn	2371

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```
SELECT pt.name, SUM(quantity) AS ordered_quantity
FROM pizzas p
    JOIN
order_details o ON o.pizza_id = p.pizza_id
    JOIN
pizza_types pt ON pt.pizza_type_id = p.pizza_type_id
GROUP BY pt.name
ORDER BY ordered_quantity DESC
LIMIT 5;
```

Result Grid		Filter Rows:
	name	ordered_quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

SELECT

```
COUNT(order_i) AS NO_ordered,  
HOUR(order_time) AS order_times
```

FROM

orders

GROUP BY order_times

Result Grid | Filter Rows:

	NO_ordered	order_times
1	1231	11
2	2520	12
3	2455	13
4	1472	14
5	1468	15
6	1920	16
7	2336	17
8	2399	18

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

SELECT

category, COUNT(name) AS pizzas

FROM

pizza_types

GROUP BY category

Result Grid |

	category	pizzas
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

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

```
select avg(quantity) from
(SELECT
    SUM(quantity) AS quantity
FROM
    orders o
    JOIN
    order_details od ON od.order_i = o.order_i
GROUP BY order_date) as order_quantity;
```

Result Grid	
	avg(quantity)
▶	138.4749

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

SELECT

```
pt.name,sum(od.quantity*p.price) as revenue  
FROM  
order_details od  
JOIN  
pizzas p ON p.pizza_id = od.pizza_id  
JOIN  
pizza_types pt ON pt.pizza_type_id = p.pizza_type_id  
group by pt.name order by revenue desc limit 3;
```

Result Grid | Filter Rows:

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
    pt.category,
    (SUM(od.quantity * p.price) / (SELECT
        SUM(order_details.quantity * pizzas.price)
    FROM
        order_details
    JOIN
        pizzas ON pizzas.pizza_id = order_details.pizza_id)) * 100 AS revenue
FROM
    order_details od JOIN
    pizzas p ON p.pizza_id = od.pizza_id JOIN
    pizza_types pt ON pt.pizza_type_id = p.pizza_type_id GROUP BY pt.category
ORDER BY revenue DESC;
```

Result Grid | Filter Rows:

	category	revenue
▶	Classic	26.905960255669903
	Supreme	25.45631126009884
	Chicken	23.955137556847493
	Veggie	23.682590927384783

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

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 Hawalian Pizza	32273.25
	The Pepperoni Pizza	30161.75

Conclusion:

- **Understanding Customer Behavior:** By analyzing order data, we gained insights into customer preferences, such as the most common pizza sizes and the top 5 most ordered pizza types. This understanding of customer behavior enables us to tailor our product offerings to better meet customer demands and preferences.
- **Optimizing Product Mix:** The identification of the highest-priced pizza and the top 3 most ordered pizza types based on revenue allows us to optimize our product mix. By focusing on popular and profitable pizza varieties, we can enhance revenue generation and profitability.
- **Operational Efficiency:** The distribution of orders by hour of the day revealed peak ordering times, enabling us to streamline operations and allocate resources more effectively to meet demand during peak periods. This optimization of operational efficiency enhances customer satisfaction by ensuring timely delivery and service.
- **Continuous Improvement:** Finally, this project underscores the importance of data-driven decision-making and continuous improvement. By regularly analyzing sales data and customer feedback, we can adapt to changing market dynamics, innovate our offerings, and stay ahead of competitors.