

PIZZA SALES ANALYSIS

- ENHANCING PRODUCT SALES STRATEGY WITH SQL



Prepared by SARBANI RAY





INTRODUCTION

The primary objective is to analyze pizza sales data using SQL and gain actionable insights into revenue, sales performance, customer preferences of pizza sales.

This enables stakeholders to make data-driven business decisions that support the company's growth.





DATASET OVERVIEW

The Pizza Sales dataset comprises of 4 key files

- Order_details.csv : Order specific data like pizza IDs and quantity.
- Orders.csv: Order level information like order dates and order times.
- Pizza_types.csv : Describing pizza categories and their ingredients.
- Pizzas.csv : Providing details of individual pizza like size ,price etc.

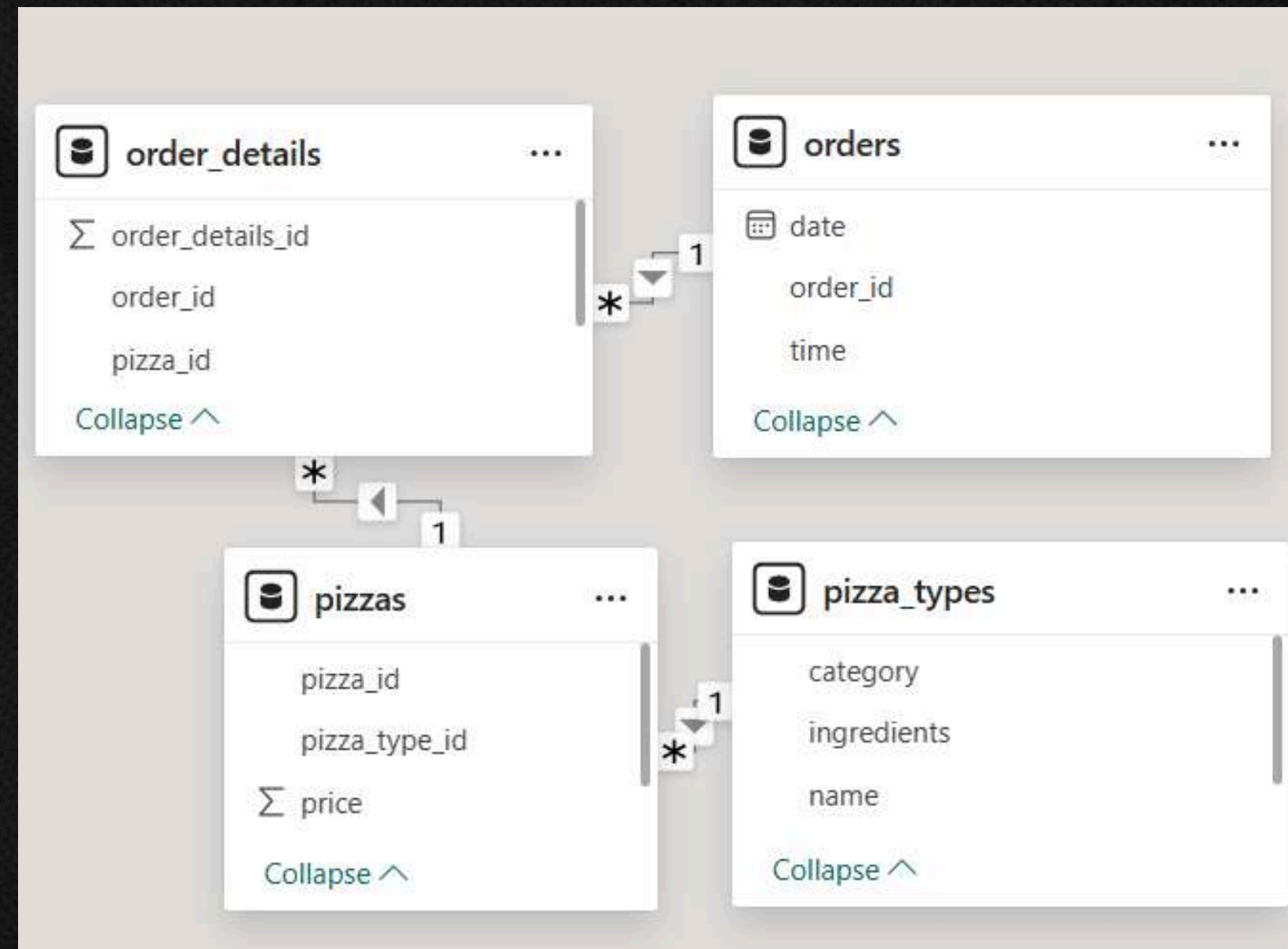


Together these CSV files enable us to analyse the order patterns of pizzas and the revenue insights and their distribution helping in gaining SQL based business insights.



DATABASE STRUCTURE OVERVIEW

This database effectively provides the necessary structure to analyse the pizza sales and customer preferences . It provides a simple ER diagram showing the relationships between these four tables.





METHODOLOGY OF ANALYSIS

01.RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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

Result Grid	
	total_orders
▶	21350

02. CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
3 • SELECT
4     ROUND(SUM(order_details.quantity * pizzas.price),
5           2) AS total_sales
6 FROM
7     order_details
8     JOIN
9     pizzas ON pizzas.pizza_id = order_details.pizza_id
```

Result Grid	
	total_sales
▶	817860.05

03 . IDENTIFY THE HIGHEST-PRICED PIZZA.

```
3  SELECT
4      pizza_types.name, pizzas.price
5  FROM
6      pizza_types
7      JOIN
8      pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9  ORDER BY pizzas.price DESC
10 LIMIT 1;
```

Result Grid			Filter Rows
	name	price	
▶	The Greek Pizza	35.95	

04 . IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
3 • SELECT
4     pizzas.size,
5     COUNT(order_details.order_details_id) AS order_count
6 FROM
7     pizzas
8     JOIN
9     order_details ON pizzas.pizza_id = order_details.pizza_id
10 GROUP BY pizzas.size
11 ORDER BY order_count DESC;
12
```

Result Grid			Filter
	size	order_count	
▶	L	18526	
	M	15385	
	S	14137	
	XL	544	
	XXL	28	

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

```
2 • SELECT
3     pizza_types.name, SUM(order_details.quantity) AS quantity
4 FROM
5     pizza_types
6     JOIN
7     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
8     JOIN
9     order_details ON order_details.pizza_id = pizzas.pizza_id
10 GROUP BY pizza_types.name
11 ORDER BY quantity DESC
12 LIMIT 5;
```

Result Grid   Filter Rows: <input type="text"/>		
	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

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

```
2
3 • SELECT
4     pizza_types.category,
5     SUM(order_details.quantity) AS quantity
6 FROM
7     pizza_types
8     JOIN
9     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
10    JOIN
11    order_details ON order_details.pizza_id = pizzas.pizza_id
12 GROUP BY pizza_types.category
13 ORDER BY quantity DESC;
```

Result Grid			Filter Rows:
	category	quantity	
•	Classic	14888	
	Supreme	11987	
	Veggie	11649	
	Chicken	11050	

07 . DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
2
3 • SELECT
4     COUNT(order_id), HOUR(order_time)
5 FROM
6     orders
7 GROUP BY HOUR(order_time)
8 ORDER BY COUNT(order_id) DESC;
```

Result Grid			Filter Rows:
	COUNT(order_id)	HOUR(order_time)	
▶	2520	12	
	2455	13	
	2399	18	
	2336	17	
	2009	19	
	1920	16	
	1642	20	
	1472	14	
	1468	15	
	1231	11	
	1198	21	
	663	22	
	28	23	
	8	10	
	1	9	

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

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

Result Grid			Filter Rows
	category	COUNT(name)	
▶	Chicken	6	
	Classic	8	
	Supreme	9	
	Veggie	9	

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

```
2
3 • SELECT
4     ROUND(AVG(quantity), 0) as avg_pizza_ordered_per_day
5 FROM
6     (SELECT
7         orders.order_date, SUM(order_details.quantity) AS quantity
8     FROM
9         orders
10    JOIN order_details ON orders.order_id = order_details.order_id
11   GROUP BY orders.order_date) AS order_quantity;
```

Result Grid		Filter Rows:
	avg_pizza_ordered_per_day	
▶	138	

10. 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 pizzas.pizza_type_id = pizza_types.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;
```

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

11 . CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

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

Result Grid			Filter
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

12 . 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
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006
2015-01-19	43365.75000000001

13. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
select category,name,revenue,SERIAL_NO
FROM(
select category,name,revenue,
dense_rank() over (partition by category
order by revenue DESC) as SERIAL_NO
from(
select c.category,c.name,
round(sum(A.quantity*b.price)) as revenue
from order_details A join pizzas B
on A.pizza_id=B.pizza_id
join pizza_types C
on B.pizza_type_id=C.pizza_type_id
group by C.category,C.name
order by C.category
) as table_A
) as table_B
where SERIAL_NO<4;
```

category	name	revenue
Chicken	The Thai Chicken Pizza	43434
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41410
Classic	The Classic Deluxe Pizza	38180
Classic	The Hawaiian Pizza	32273
Classic	The Pepperoni Pizza	30162
Supreme	The Spicy Italian Pizza	34831
Supreme	The Italian Supreme Pizza	33477
Supreme	The Sicilian Pizza	30940
Veggie	The Four Cheese Pizza	32266
Veggie	The Mexicana Pizza	26781
Veggie	The Five Cheese Pizza	26066

BUSINESS INSIGHTS & RECOMMENDATIONS

- **Popular Sizes(L & M)** : Large Pizzas are the most commonly ordered size pizza indicating larger size preference among the customers followed by M .
- **Top Performer** : The Classic Deluxe Pizza is the most ordered (quantity sold) pizza type followed by others . Consistent availability of the ingredients for this top selling pizza should be ensured.
- Classic Category has the highest total quantity of pizza ordered in comparison to Supreme , Veggie and Chicken.
- **Peak Hours** : Peak hours for the orders are in the afternoon (12-1 PM) and in the evening (5 PM -7 PM).We can optimise staffing during these peak hours for efficient service.
- **High Revenue Categories** : In terms of Revenue , the "Classic" & "Supreme" categories dominated . Marketing efforts can be increased by focussing on these two categories to earn more profit .
- Average pizza ordered per day is 138 .
- **Financial Performance** : Total Revenue generated from pizza sales is around \$ 8 ,17,860 out of 21 ,350 orders.
- **Customer Feedback and Ratings** : By regularly gathering feedbacks and ratings , continuous improvement can be done.



THANK YOU!

