



PIZZAS SALES ANALYSIS

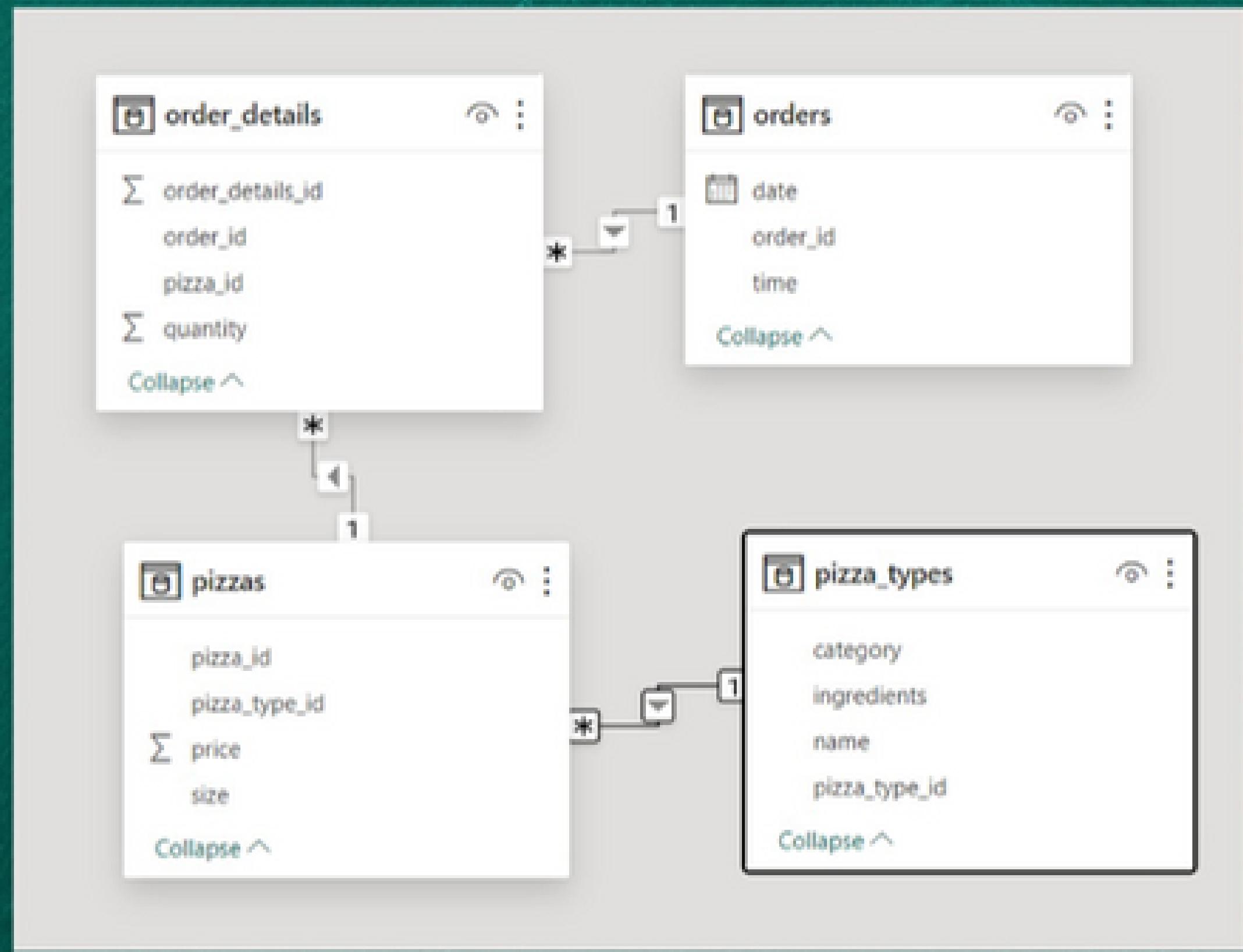
BY SQL

PROJECT STATEMENT

Hello, I'm Hardik Sharma. Using SQL queries, I'll analyze pizza sales data to uncover trends and insights. We'll explore sales trends, popular pizzas, customer preferences, and more. Let's harness the power of data to drive informed decisions and optimize our business strategies. Excited to collaborate!



DATA MODEL VIEW



1. Retrieve the total number of orders placed.

QUERY

SELECT

```
COUNT(order_id) AS TOTAL_ORDERS
```

FROM

```
pizza_sales_project.orders
```

OUTPUT

Result Grid

TOTAL_ORDERS
21350

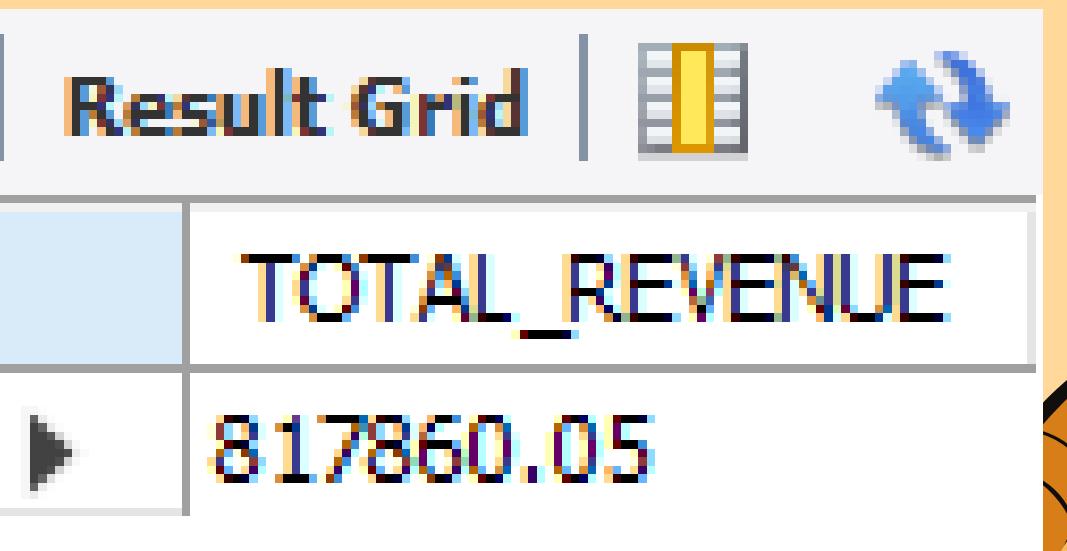


2. Calculate the total revenue generated from pizza sales.

QUERY

```
SELECT  
    ROUND(SUM(Q.quantity * P.price), 2) AS TOTAL_REVENUE  
FROM  
    pizzas P  
        JOIN  
    orders_detail Q ON P.pizza_id = Q.pizza_id
```

OUTPUT



	TOTAL_REVENUE
▶	817860.05

3. Identify the highest-priced pizza.

QUERY

```
SELECT  
    P1.name, P2.price  
FROM  
    pizza_types P1  
        JOIN  
    pizzas P2 ON P1.pizza_type_id = P2.pizza_type_id  
ORDER BY P2.price DESC  
LIMIT 1
```

OUTPUT

Result Grid | Filter Row |

	name	price
▶	The Greek Pizza	35.95

4. Identify the most common pizza size ordered.

QUERY

```
SELECT  
    P1.size, COUNT(01.order_detail_id) AS TOTAL_COUNT  
FROM  
    pizzas P1  
    JOIN  
    orders_detail 01 ON P1.pizza_id = 01.pizza_id  
GROUP BY P1.size  
ORDER BY TOTAL_COUNT DESC
```

OUTPUT

	size	TOTAL_COUNT
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

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

QUERY

```
SELECT  
    P1.name, SUM(01.quantity) AS TOTAL_ORDER  
FROM  
    pizza_types P1  
        JOIN  
    pizzas P2 ON P1.pizza_type_id = P2.pizza_type_id  
        JOIN  
    orders_detail 01 ON 01.pizza_id = P2.pizza_id  
GROUP BY P1.name  
ORDER BY TOTAL_ORDER DESC  
LIMIT 5
```

OUTPUT

	name	TOTAL_ORDER
▶	The Classic Deluxe Pizza	2453
▶	The Barbecue Chicken Pizza	2432
▶	The Hawaiian Pizza	2422
▶	The Pepperoni Pizza	2418
▶	The Thai Chicken Pizza	2371



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

QUERY

```
SELECT  
    P2.category, COUNT(P2.category)  
FROM  
    orders_detail 01  
        JOIN  
    pizzas P1 ON P1.pizza_id = 01.pizza_id  
        JOIN  
    pizza_types P2 ON P2.pizza_type_id = P1.pizza_type_id  
GROUP BY P2.category
```

OUTPUT

	category	COUNT(P2.category)
▶	Classic	14579
	Veggie	11449
	Supreme	11777
	Chicken	10815



7. Determine the distribution of orders by hour of the day.

Result Grid

	Hours	orders
▶	9	1
	10	8
	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

QUERY

```
SELECT HOUR(order_time) AS Hours, COUNT(order_id) AS orders
FROM orders
GROUP BY HOUR(order_time)
ORDER BY Hours
```

<<<OUTPUT

8. Find the category-wise distribution of pizzas.

QUERY

```
SELECT  
    category, COUNT(category) AS count  
FROM  
    pizza_types  
GROUP BY category
```

OUTPUT

	category	count
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

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

SELECT

```
ROUND( AVG(Total_quantity), 0) AS Days_avg
```

FROM

(SELECT

```
    01.order_date, SUM(02.quantity) AS Total_quantity
```

FROM

```
    orders 01
```

```
JOIN orders_detail 02 ON 02.order_id = 01.order_id
```

```
GROUP BY 01.order_date) AS temp
```

OUTPUT

Result Grid

	Days_avg
▶	138

QUERY

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

QUERY

```
SELECT P2.name, SUM(01.quantity * P1.price) AS REVENUE  
FROM orders_detail 01  
JOIN pizzas P1 ON 01.pizza_id = P1.pizza_id  
JOIN pizza_types P2 ON P2.pizza_type_id = P1.pizza_type_id  
GROUP BY P2.name  
ORDER BY REVENUE DESC  
LIMIT 3
```

OUTPUT

Result Grid		Filter Rows:
	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.

QUERY

```
SELECT P2.category, ROUND((SUM(01.quantity * P1.price)/(SELECT  
ROUND(SUM(Q.quantity * P.price), 2) AS TOTAL_REVENUE  
FROM pizzas P  
JOIN orders_detail Q ON P.pizza_id = Q.pizza_id )*100),2) AS REVENUE_PERCENTS  
FROM orders_detail 01  
JOIN pizzas P1 ON 01.pizza_id = P1.pizza_id  
JOIN pizza_types P2 ON P2.pizza_type_id = P1.pizza_type_id  
GROUP BY P2.category
```

OUTPUT

	category	REVENUE_PERCENTS
▶	Classic	26.91
	Veggie	23.68
	Supreme	25.46
	Chicken	23.96

THANK YOU

Project by-
Hardik Sharma

Email - hardik24sharma99@gmail.com