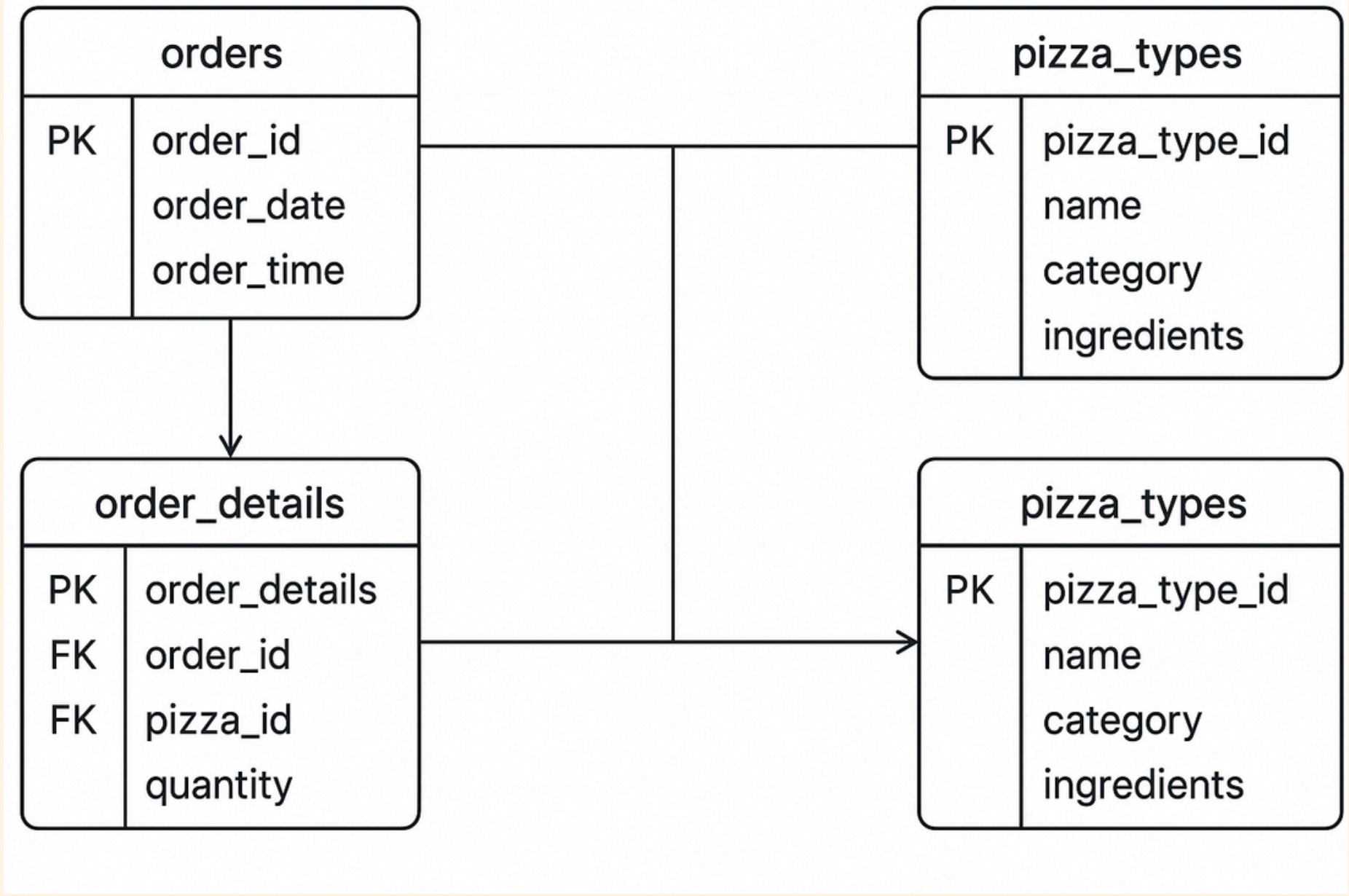
I'M VIVEK SHARMA, A DATA ENTHUSIAST.

SQL Project Pizza Sales Analysis

This project "Pizza Sales Analysis" is based on SQL, where I analyzed pizza sales data to find key insights like top-selling pizzas, total revenue, and customer preferences.

Schema Diagram



Q1. Retrieve the total number of orders placed

```sql
-- Retrieve the total number of orders placed

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

Result Grid

	Total_Orders
▶	21350

Q2. Calculate the total revenue generated from pizza sales

```sql
-- 2 Calculate the total revenue generated from pizza sales

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

Don't Limit

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

Total_Revenue
817860.05

Q . Identify the highest-priced pizza

```sql
-- 3 Identify the highest-priced pizza

SELECT
    pizza_id,
    price AS Highest_Price
FROM pizzas
ORDER BY price DESC
LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap

pizza_id	Highest_Price
the_greek_xxl	35.95

Q . Identify the most common pizza size ordered

```sql
-- 4 Identify the most common pizza size ordered

SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS Total_Orders
FROM orders_details
JOIN pizzas
    ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizzas.size
ORDER BY Total_Orders DESC
LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

size	Total_Orders
L	18526

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

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Pizza_Name	Total_Quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Q6 .Join the necessary tables to find the total quantity of each pizza category ordered

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	Total_Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

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

```sql
SELECT
    HOUR(order_time) AS Hour,
    COUNT(order_id) AS Total_Orders
FROM orders
GROUP BY HOUR(order_time)
ORDER BY Hour;
```

Result Grid | Filter Rows: | Export:

Hour	Total_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

Q 8. Join relevant tables to find the category-wise distribution of pizzas

```sql
SELECT
    pizza_types.category,
    COUNT(DISTINCT pizzas.pizza_id) AS Total_Pizza_Types
FROM pizzas
JOIN pizza_types
    ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	Total_Pizza_Types
Chicken	18
Classic	26
Supreme	25
Veggie	27

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

```sql
SELECT
    ROUND(AVG(Daily_Pizza_Count), 2) AS Avg_Pizzas_Per_Day
FROM (
    SELECT
        orders.order_date,
        SUM(orders_details.quantity) AS Daily_Pizza_Count
    FROM orders
    JOIN orders_details
        ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date
) AS DailyStats;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Avg_Pizzas_Per_Day
138.47

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

```sql
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * price) AS revenue
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 revenue DESC
LIMIT 3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

Q11. Calculate the percentage contribution of each pizza type to total revenue.

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

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Q 12. Analyze the cummulative 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 orders.order_date
) AS sales;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

This SQL project on Pizza Sales Analysis helped in understanding how to use data to gain business insights.
By using SQL queries, I analyzed sales trends, top-performing pizzas, and revenue patterns.
It enhanced my knowledge of database design, schema creation, and data analysis — giving me a strong foundation for real-world data analytics projects.

Hope you liked it! Thank you