Data-Driven Insights for Business Growth

PIZZA SALES SQL PROJECT

By ARUHI PALLAVI

INTRODUCTION

- This SQL project is built using real-world pizza sales data from a pizza store.
- **Our goal:** Extract insights to boost revenue, optimize operations, and align marketing with customer behavior.
- We explored sales trends, top performers, peak hours, and customer preferences.

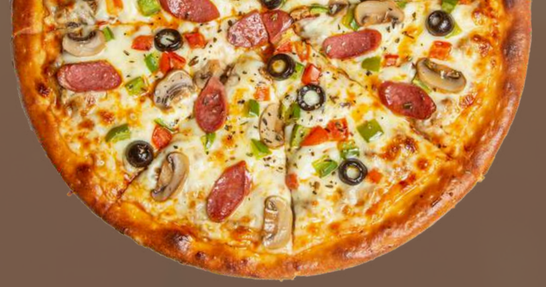
DATASET OVERVIEW

THE DATASET INCLUDES 4 TABLES:

- **Orders:** Contains order date and time.
- **Order Details:** Tracks which pizzas were ordered and in what quantity.

- **PIZZAS:** PRICE AND SIZE OF EACH PIZZA.
- **PIZZA TYPES:** PIZZA CATEGORY (VEGGIE, CLASSIC, ETC.)

ANALYSIS



1. Retrieve the total number of orders placed.

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

Results | Messages

	total_orders
1	21350

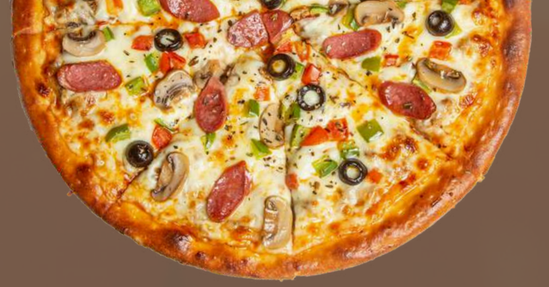
2. Calculate the total revenue generated from pizza sales.

```
SELECT
SUM(od.quantity * p.price) AS total_revenue
FROM order_details od
JOIN pizzas p
ON od.pizza_id = p.pizza_id;
```

Results | Messages

	total_revenue
1	817860.05

ANALYSIS



3. Identify the highest-priced pizza.

```sql
SELECT
TOP 1 pizza_id, price
FROM pizzas
ORDER BY price DESC;
```

Results | Messages

	pizza_id	price
1	the_greek_xxl	35.95

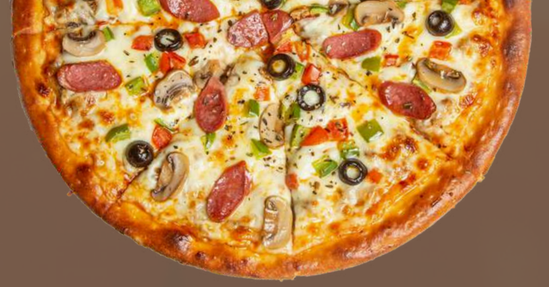
4. Identify the most common pizza size ordered.

```sql
SELECT
TOP 1 p.size, COUNT(*) AS count
FROM order_details od
JOIN pizzas p
ON od.pizza_id = p.pizza_id
GROUP BY p.size
ORDER BY count DESC;
```

Results | Messages

	size	count
1	L	18526

ANALYSIS



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

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

Results | Messages

	name	total_quantity
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

6. find the total quantity of each pizza category ordered.

```
SELECT
pt.category, SUM(od.quantity) AS total_quantity
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.category;
```

Results | Messages

	category	total_quantity
1	Mushroom	1359
2	Chicken	11050
3	Classic	13529
4	Supreme	11987
5	Veggie	11649

ANALYSIS

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

```
SELECT
DATEPART(HOUR, CAST(order_time AS TIME)) AS order_hour,
COUNT(*) AS total_orders
FROM orders
GROUP BY DATEPART(HOUR, CAST(order_time AS TIME))
ORDER BY order_hour;
```

Results | Messages

	order_hour	total_orders
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28

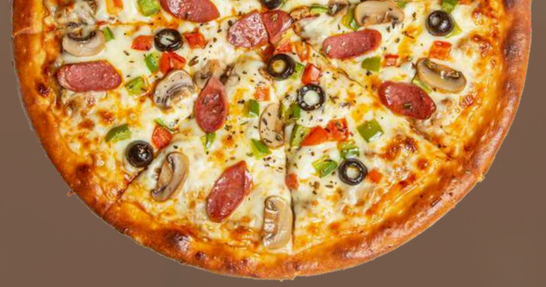
8. find the category-wise distribution of pizzas.

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

Results | Messages

	category	pizza_count
1	Mushroom	3
2	Chicken	18
3	Classic	23
4	Supreme	25
5	Veggie	27

ANALYSIS



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

```sql
SELECT AVG(pizzas_per_day) AS avg_pizzas_ordered
FROM (
  SELECT o.order_date, SUM(od.quantity) AS pizzas_per_day
  FROM orders o
  JOIN order_details od ON o.order_id = od.order_id
  GROUP BY o.order_date
  ) AS daily_totals;
```

Results | Messages

	avg_pizzas_ordered
1	138

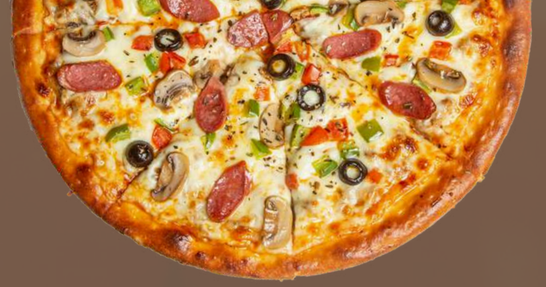
10. Determine the top 3 most ordered pizza types based on revenue.

```sql
SELECT TOP 3 pt.name, SUM(od.quantity * p.price) AS revenue
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.name
ORDER BY revenue DESC;
```

Results | Messages

	name	revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768.00
3	The California Chicken Pizza	41409.50

ANALYSIS



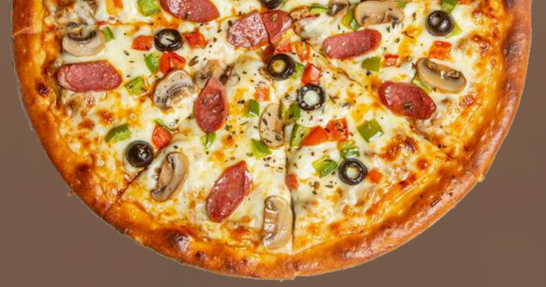
11. Calculate the percentage contribution of each pizza type to total revenue.

```sql
WITH revenue_by_type AS (
    SELECT pt.name, SUM(od.quantity * p.price) AS revenue
    FROM order_details od
    JOIN pizzas p ON od.pizza_id = p.pizza_id
    JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
    GROUP BY pt.name
 ),
total AS (
    SELECT SUM(revenue) AS total_revenue FROM revenue_by_type
 )
SELECT rbt.name,
       rbt.revenue,
       ROUND((rbt.revenue / t.total_revenue) * 100, 2) AS percentage
FROM revenue_by_type rbt, total t
ORDER BY percentage DESC;
```

Results | Messages

	name	revenue	percentage
1	The Thai Chicken Pizza	43434.25	5.310000
2	The Barbecue Chicken Pizza	42768.00	5.230000
3	The California Chicken Pizza	41409.50	5.060000
4	The Classic Deluxe Pizza	38180.50	4.670000
5	The Spicy Italian Pizza	34831.25	4.260000
6	The Southwest Chicken Pizza	34705.75	4.240000
7	The Italian Supreme Pizza	33476.75	4.090000
8	The Four Cheese Pizza	32265.70	3.950000
9	The Hawaiian Pizza	32273.25	3.950000
10	The Sicilian Pizza	30940.50	3.780000
11	The Pepperoni Pizza	30161.75	3.690000
12	The Greek Pizza	28454.10	3.480000
13	The Mexicana Pizza	26780.75	3.270000
14	The Five Cheese Pizza	26066.50	3.190000
15	The Pepper Salami Pizza	25529.00	3.120000
16	The Italian Capocollo Pizza	25094.00	3.070000

ANALYSIS



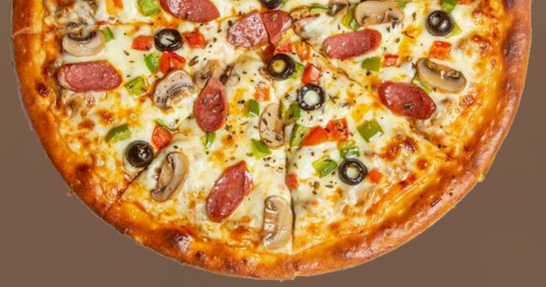
12. Analyze the cumulative revenue generated over time.

```sql
SELECT o.order_date,
       SUM(od.quantity * p.price) AS daily_revenue,
       SUM(SUM(od.quantity * p.price))
       OVER (ORDER BY o.order_date)AS cumulative_revenue
FROM orders o
JOIN order_details od ON o.order_id = od.order_id
JOIN pizzas p ON od.pizza_id = p.pizza_id
GROUP BY o.order_date
ORDER BY o.order_date;
```

Results | Messages

	order_date	daily_revenue	cumulative_revenue
1	2015-01-01	2713.85	2713.85
2	2015-01-02	2731.90	5445.75
3	2015-01-03	2662.40	8108.15
4	2015-01-04	1755.45	9863.60
5	2015-01-05	2065.95	11929.55
6	2015-01-06	2428.95	14358.50
7	2015-01-07	2202.20	16560.70
8	2015-01-08	2838.35	19399.05
9	2015-01-09	2127.35	21526.40
10	2015-01-10	2463.95	23990.35
11	2015-01-11	1872.30	25862.65
12	2015-01-12	1919.05	27781.70
13	2015-01-13	2049.60	29831.30
14	2015-01-14	2527.40	32358.70
15	2015-01-15	1984.80	34343.50

ANALYSIS



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

```sql
WITH category_revenue AS (
    SELECT pt.category, pt.name AS pizza_name,
    SUM(od.quantity * p.price) AS revenue
    FROM order_details od
    JOIN pizzas p ON od.pizza_id = p.pizza_id
    JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
    GROUP BY pt.category, pt.name
),
ranked AS (
    SELECT *,
           RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS rnk
    FROM category_revenue
)
SELECT category, pizza_name, revenue
FROM ranked
WHERE rnk <= 3;
```

Results | Messages

	category	pizza_name	revenue
1	Mushroom	"The Pepperoni	18834.50
2	Chicken	The Thai Chicken Pizza	43434.25
3	Chicken	The Barbecue Chicken Pizza	42768.00
4	Chicken	The California Chicken Pizza	41409.50
5	Classic	The Classic Deluxe Pizza	38180.50
6	Classic	The Hawaiian Pizza	32273.25
7	Classic	The Pepperoni Pizza	30161.75
8	Supreme	The Spicy Italian Pizza	34831.25
9	Supreme	The Italian Supreme Pizza	33476.75
10	Supreme	The Sicilian Pizza	30940.50
11	Veggie	The Four Cheese Pizza	32265.70
12	Veggie	The Mexicana Pizza	26780.75
13	Veggie	The Five Cheese Pizza	26066.50

KEY INSIGHTS

Total Orders: 21,350

↪ Indicates a healthy order volume and customer reach.

Total Revenue: $817,860.05

↪ Reflects strong market potential.

Most Popular Size: Large (L) – 18,526 orders

↪ Customers favor larger portions — opportunity for combo deals.

Average Daily Orders: 138

↪ Helps in inventory and manpower planning.

ORDER BEHAVIOR PATTERNS

Peak Hours:

- 12 PM – 3 PM & 6 PM – 9 PM
- Focus ad campaigns and staff allocation here.

Top Pizza Categories (by Quantity):

- Classic and Supreme lead, suggesting traditional preferences dominate.

Top Revenue-Generating Pizza:

- Thai Chicken Pizza — ideal for upselling and promotional spotlight.

ADVANCED OBSERVATIONS

Top-selling pizzas contribute 5–6% each to revenue

🔁 Healthy distribution, business isn’t dependent on just one item.

Cumulative Revenue Trend:

🔁 Stable and growing — reflects consistent demand.

Least Varied Category: Mushroom

🔁 A potential gap — explore new offerings in this niche.

BUSINESS TAKEAWAYS

Prioritize stocking and marketing Classic & Chicken pizzas.

Leverage L/XL sizes in combo offers — match demand.

Align staffing and offers to peak ordering hours.

Innovate low-performing categories (like Mushroom) to expand market.

Use insights to shape seasonal promotions and upselling strategies.

Thank you for Attention

ARUHI PALLAVI

+91 9608880418

Pallaviaruhi@gmail.com

Greater Noida, 201310