SQL PROJECT ON PIZZA SALES

PROJECT DESCRIPTION

This project analyzes a pizza sales dataset using SQL to extract meaningful business insights. The objective was to understand sales performance, customer ordering patterns, revenue trends, and top-performing pizza categories.

KEY ASPECTS

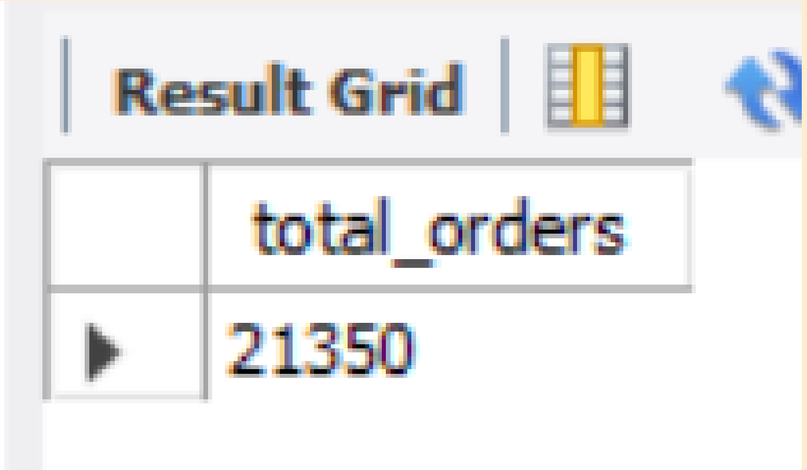
- Analyze total revenue and sales performance
- Identify top-selling pizzas and categories
- Determine peak order hours and busiest days
- Calculate average daily sales
- Generate cumulative revenue trends

INSIGHT GENERATED

- Identified top 3 pizzas in each category based on revenue
- Found peak ordering hours of the day
- Calculated category-wise revenue contribution
- Analyzed cumulative revenue growth over time
- Determined average pizzas sold per day

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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



CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price), 2) AS total_sales
FROM order_details
JOIN pizzas
```

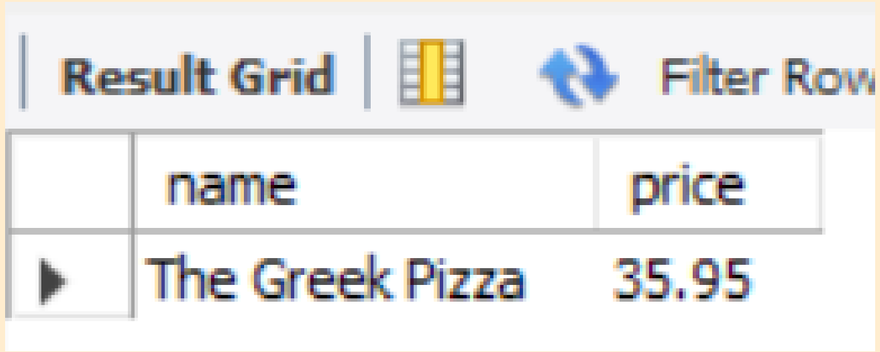
Result Grid

total_sales
676559.6

IDENTIFY THE HIGHEST PRICED PIZZA.

```sql
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM pizzas
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
```

```sql
GROUP BY pizzas.size
ORDER BY order_count DESC;
```



	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM pizzas
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
```

```
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid | Filter

size	order_count
L	15338
M	12736
S	11655
XL	456
XXL	24

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

```
SELECT
    pizza_types.name,
    SUM(order_details.quantity) AS quantity
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.name
ORDER BY quantity DESC
LIMIT 5;
```

Result Grid | Filter Rows:

name	quantity
The Barbecue Chicken Pizza	6066
The Spicy Italian Pizza	4767
The Four Cheese Pizza	4734
The Pepperoni Pizza	4044
The Classic Deluxe Pizza	4024

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

```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
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
  ORDER BY quantity DESC;
```

Result Grid | Filter Rows

category	quantity
Veggie	26467
Supreme	18630
Chicken	15081
Classic	12332
southw_ckn_l	3179

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
select
hour(order_time) as hour , count(order_id) as order_count
From
orders
group by hour(order_time);
```

Result Grid | Filter Rows:

hour	order_count
20	1642
21	1198
22	663
23	28
10	8
9	1

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

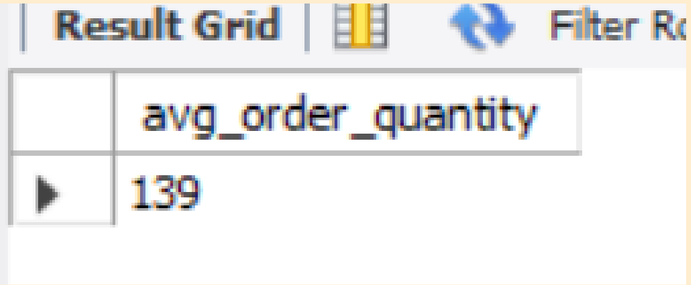
```
select category , count(name) from pizza_types
group by category;
```

Result Grid | Filter Rows:

category	count(name)
ital_cpclo_m	2
southw_ckn_l	2
cali_ckn_m	2
pepperoni_s	1
peppr_salami_m	1
thai_ckn_l	1

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

```
SELECT
    ROUND(AVG(quantity), 0) AS avg_order_quantity
FROM (
    SELECT
        orders.order_date,
        SUM(order_details.quantity) AS quantity
    FROM orders
    JOIN order_details
        ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date
) AS order_quantity;
```



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

```sql
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;
```

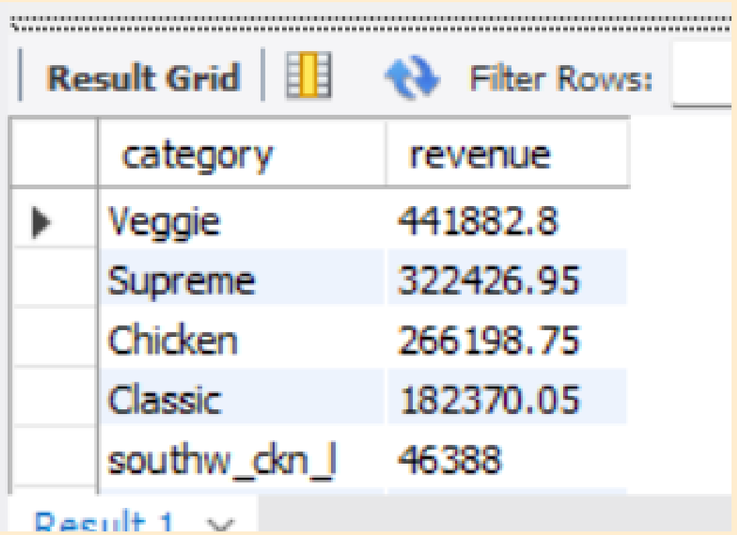
Result Grid | Filter Rows: | Export:

name	revenue
The Barbecue Chicken Pizza	106609.5
The Spicy Italian Pizza	86549.25
The Four Cheese Pizza	80271.29999999524

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

```sql
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price), 2) AS revenue
FROM pizza_types
JOIN pizzas
    ON pizza_types.pizza_type_id = pizzas.pizza_type_id
```

```sql
JOIN order_details
    ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```



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;
```

Result Grid | Filter Rows:

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

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

```sql
SELECT name, revenue
FROM (SELECT category,
        name,
        revenue,
        RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS rn
    FROM (
```

```sql
            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;
```

```sql
SELECT
    pizza_types.category,
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM pizza_types
JOIN pizzas
```

Result Grid | Filter Rows: | Export:

name	revenue
The Sicilian Pizza	50318.5
The Pepperoni Pizza	25231
17731	28802
The Four Cheese Pizza	80271.29999999524
The Five Cheese Pizza	66544.5

PROJECT OUTCOME

The project helped in understanding how SQL can be used for real-world business analytics and decision-making in the food retail industry.


THANK YOU