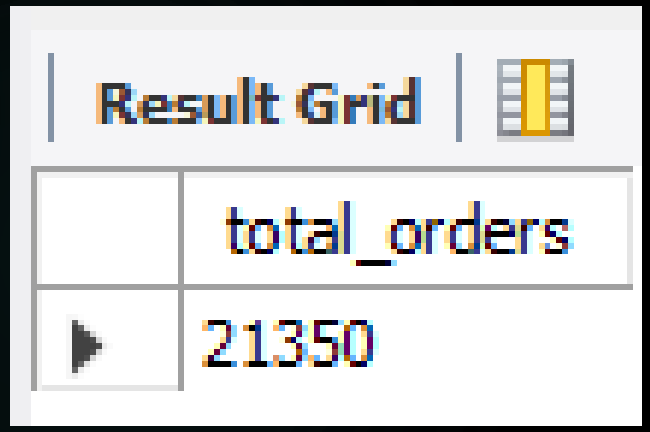
Comprehensive Pizza Sales Analysis

USING ADVANCED SQL QUERIES

1. Retrieve the Total Number of Orders Placed

This query calculates the total number of pizza orders placed by customers.

```
select count(order_id) as total_orders from orders;
```



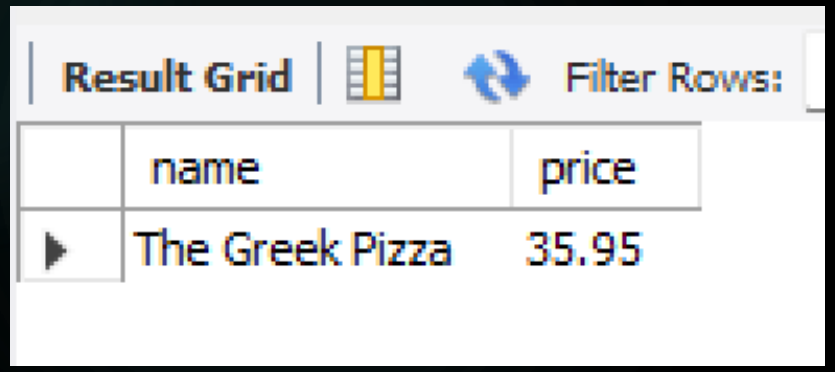
```sql
SELECT
    ROUND(SUM(orders_details.quantity * pizzas.price),
            2) AS total_revenue_generated
FROM
    orders_details
        JOIN
    pizzas ON pizzas.pizza_id = orders_details.pizza_id
```

Result Grid | Filter Rows:

total_revenue_generated
817860.05

- 2. Calculate the Total Revenue Generated from Pizza Sales
- This query determines the total revenue generated from all pizza orders by multiplying the quantity ordered by the price.

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



- 3. Identify the Highest-Priced Pizza
- This query retrieves the most expensive pizza and its price.

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

Result Grid | Filter Rows:

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

4. Identify the Most Common Pizza Size Ordered

This query finds out the most frequently ordered pizza size.

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

Result Grid | Filter Rows:

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

- 5. List the Top 5 Most Ordered Pizza Types with Quantities
- This query provides the top 5 pizza types based on the total quantity ordered.

```sql
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
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 quantity DESC;
```

Result Grid | Filter R

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

- 6. Total Quantity of Each Pizza Category Ordered
- This query summarizes the quantity of pizzas ordered by their category (e.g. Classic, Supreme, Veggie, Chicken).

Result Grid | Filter R

hour	order_count
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
10	8
9	1

```sql
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

7. Distribution of Orders by Hour of the Day

This query shows when most orders are placed, grouping by hour.

```sql
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

8. Category-wise Distribution of Pizza Orders

This query displays how pizza orders are distributed across different categories.

```sql
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid | Filter Rows:

avg_pizza_ordered_per_day
138

9. Average Number of Pizzas Ordered Per Day

This query calculates the average number of pizzas ordered daily.

```sql
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.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:

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

10. Top 3 Most Ordered Pizza Types Based on Revenue

This query lists the top 3 pizza types that generated the highest revenue.

```sql
SELECT
    pizza_types.category,
    round(SUM(orders_details.quantity * pizzas.price) / (SELECT
            ROUND(SUM(orders_details.quantity * pizzas.price),
                    2)
        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) * 100, 2) AS revenue_percentage
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_percentage DESC;
```

Result Grid | Filter Rows:

category	revenue_percentage
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

11. Percentage Contribution of Each Pizza Type to Total Revenue

This query shows the percentage of revenue each pizza category contributes to the total.

```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:

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

12. Cumulative Revenue Generated Over Time

This query shows the cumulative revenue generated as orders progress over time.

```sql
SELECT
    name,
    revenue
FROM
    (SELECT
        category,
        name,
        revenue,
        RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS rn
     FROM
        (SELECT
            pizza_types.category,
            pizza_types.name,
            SUM(orders_details.quantity * pizzas.price) 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, pizza_types.name
        ) AS a
    ) AS b
WHERE
    rn <= 3;
```

Result Grid | Filter Rows: | Exp

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

- **13. Top 3 Most Ordered Pizza Types Based on Revenue for Each Category**
- This query determines the top 3 pizza types based on revenue for each pizza category.