SALES OVERVIEW

INTRODUCTION

This analysis focuses on pizza sales using data from a pizza shop dataset. I have addressed a series of questions aimed at analyzing overall sales figures.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

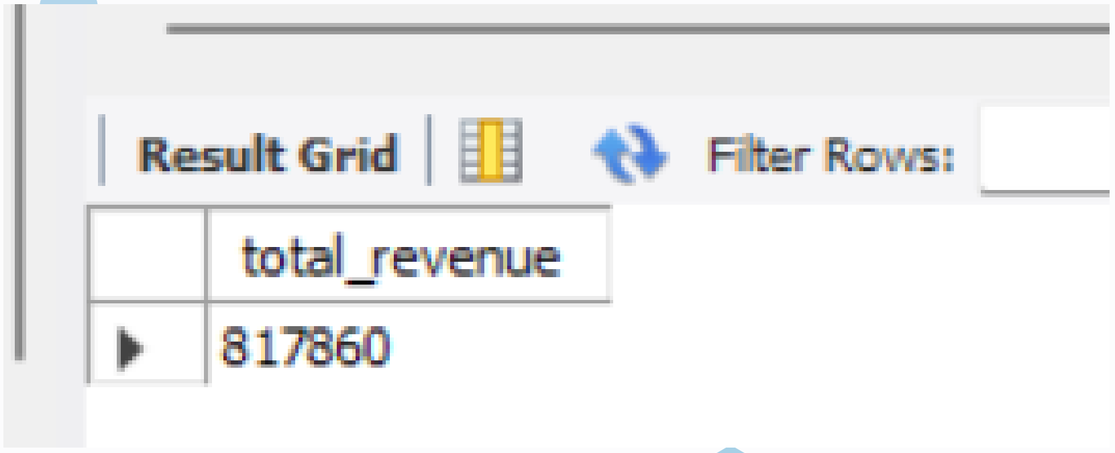
```sql
-- Retrieve the total number of orders placed.
SELECT
    COUNT(*)
FROM
    orders;
```

Result Grid | Filter Rows:

	COUNT(*)
▶	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```sql
-- Calculate the total revenue generated from pizza sales.
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price)) AS total_revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```



IDENTIFY THE HIGHEST-PRICED PIZZA.

```
2 •  SELECT
3        pizza_types.name, pizzas.price AS highest_priced_pizza
4    FROM
5        pizza.pizzas
6            JOIN
7        pizza.pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
8    ORDER BY pizzas.price DESC
9    LIMIT 1;
```

Result Grid | Filter Rows: | Expo

name	highest_priced_pizza
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```sql
-- Identify the most common pizza size ordered.
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS no_pizzas
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY no_pizzas DESC
LIMIT 1;
```

Result Grid | Filter Rows:

size	no_pizzas
L	18526

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

```sql
-- List the top 5 most ordered pizza types along with their quantities.
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quan
FROM
    pizza_types
        JOIN
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
        AND pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types.name
ORDER BY quan DESC
LIMIT 5;
```

Result Grid | Filter Rows:

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

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

```sql
SELECT
    pizza_types.category, SUM(order_details.quantity) AS quan
FROM
    pizza_types
        JOIN
    order_details
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        AND pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_types.category;
```

Result Grid

	category	quan
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
-- Determine the distribution of orders by hour of the day.
select hour(time),count(order_id)as count_orders
from orders
group by hour(time);
```

hour(time)	count_orders
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

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

```
-- Join relevant tables to find the category-wise distribution of pizzas.
select category, count(name) from pizza_types
group by category
```

category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

```sql
-- Group the orders by date and calculate the average number of pizzas ordered per day.
SELECT
    ROUND(AVG(sum_orders))
FROM
    (SELECT
        SUM(order_details.quantity) AS sum_orders, orders.date
    FROM
        order_details
    JOIN orders ON order_details.order_id = orders.order_id
    GROUP BY orders.date) AS derived;
```

	ROUND(AVG(sum_orders))
▶	138

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

```sql
-- Determine the top 3 most ordered pizza types based on revenue.
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;
```

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

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

```sql
-- Calculate the percentage contribution of each pizza type to total revenue.
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                ROUND(SUM(order_details.quantity * pizzas.price),
                        1) AS sales
            FROM
                order_details
                    JOIN
                pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
            1) AS revenue
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 revenue DESC
```

	category	revenue
▶	Classic	26.9
	Supreme	25.5
	Chicken	24
	Veggie	23.7

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```sql
Select date,round(sum(revenue) over (order by date),2)as cumulative_revenue
from
(select date, sum(order_details.quantity * pizzas.price) as revenue from order_details join pizzas
join orders
on order_details.pizza_id = pizzas.pizza_id and order_details.order_id = orders.order_id
group by date) as daily_sales;
```

date	cumulative_revenue
2015-01-01	2713.85
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	23990.35
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.3
2015-01-14	32358.7
2015-01-15	34343.5
2015-01-16	36937.65
2015-01-17	39001.75
2015-01-18	40978.6
2015-01-19	43365.75
2015-01-20	45763.65
2015-01-21	47804.2
2015-01-22	50300.9
2015-01-23	52724.6
2015-01-24	55013.85
2015-01-25	56631.4
2015-01-26	58515.8
2015-01-27	61043.85
2015-01-28	63059.85
2015-01-29	65105.15
2015-01-30	67375.45
2015-01-31	69793.3
2015-02-01	72982.5
2015-02-02	75311.1
2015-02-03	77925.9
2015-02-04	80159.8
2015-02-05	82375.6
2015-02-06	84885.55
2015-02-07	87123.2
2015-02-08	89158.2
2015-02-09	91353.55
2015-02-10	93410.05
2015-02-11	95870.05
2015-02-12	98028.85
2015-02-13	100783.35
2015-02-14	103102.5
2015-02-15	105243.75

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

```sql
-- Determine the top 3 most ordered pizza types based on revenue for each pizza category.
select category , name, revenue from
(select category , name, revenue, rank() over(partition by category order by revenue desc) as rnk
from
(Select pizza_types.category , pizza_types.name , sum(order_details.quantity * pizzas.price) as revenue
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 , pizza_types.name) as in_table)as out_table
where rnk<=3 ;
```

category	name	revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5
Classic	The Classic Deluxe Pizza	38180.5
Classic	The Hawaiian Pizza	32273.25
Classic	The Pepperoni Pizza	30161.75
Supreme	The Spicy Italian Pizza	34831.25
Supreme	The Italian Supreme Pizza	33476.75
Supreme	The Sicilian Pizza	30940.5
Veggie	The Four Cheese Pizza	32265.700000000
Veggie	The Mexicana Pizza	26780.75
Veggie	The Five Cheese Pizza	26066.5

"RETURN FOR MORE ADVENTURES!"