PRESENTED BY AMAN

PIZZA SALES ANALYSIS USING SQL

- WHERE EVERY SLICE TELLS A STORY

OBJECTIVE

TO ANALYZE PIZZA SALES DATA USING SQL TO UNCOVER KEY INSIGHTS SUCH AS:

- Best-selling pizza types
- Revenue trends
- Customer order patterns
- Inventory and sales performance

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.


```
SELECT
    COUNT(order_details_id) AS total_order
FROM
    order_details;
```

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

total_order
64748

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),
            2) AS total_revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

Result Grid

	total_revenue
▶	1089121.35

IDENTIFY THE HIGHEST-PRICED PIZZA.

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

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

Result Grid | Filter Rows:

	pizza	price
▶	the_greek_xxl	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 order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC
LIMIT 1;
```

Result Grid | Filter

size	order_count
L	24684

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 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	3273
The Hawaiian Pizza	3232
The Classic Deluxe Pizza	3227
The Pepperoni Pizza	3210
The Thai Chicken Pizza	3145

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 order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC
LIMIT 1;
```

Result Grid | Filter

size	order_count
L	24684

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
-- Determine the distribution of orders by hour of the day.


SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid | Filter Rows:

hour(order_time)	count(order_id)
11	1231
12	2520
13	2455
14	1472
15	1468

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;
```

Result Grid | Filter Rows:

hour(order_time)	count(order_id)
11	1231
12	2520
13	2455
14	1472
15	1468

SHODWE

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

Contact

Execute the selected portion of the script or everything, if there is no selection

```sql
SELECT
    ROUND(AVG(quantity), 0) as avg_quant_day
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;
```

Result Grid

	avg_quant_day
▶	184

SHODWE

Contact

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

```
select name, revenue
from
(select category, name,revenue,
rank() over(partition by category order by revenue desc) as Ran
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 a) as b
where Ran<=3;
```

Result Grid | Filter Rows:

name	revenue
The Barbecue Chicken Pizza	57630.75
The Thai Chicken Pizza	57542.75
The California Chicken Pizza	54875.75
The Clas The Classic Deluxe Pizza	
The Hawaiian Pizza	42977.25

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),
                                2) AS total_revenue
                FROM
                    order_details
                        JOIN
                    pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100),
            2) 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;
```

Result Grid | Filter Row

category	revenue
Classic	26.82
Supreme	25.45
Chicken	23.96
Veggie	23.77

THANK YOU

FOR ATTENTION

2025 PIZZA RESTO PRESENTATION