PIZZA SALES ANALYSIS USING SQL

ABOUT PROJECT

The "Pizza Sales Analysis" project is designed to explore key insights and patterns from pizza sales data, leveraging SQL queries and analytics techniques. This project addresses several critical questions related to sales and customer preferences, such as:

1. Total Orders: Retrieving the total number of orders placed by customers.
2. Revenue Analysis: Calculating the total revenue generated from all pizza sales.
3. Pricing Insights: Identifying the highest-priced pizza on the menu.
4. Order Preferences: Determining the most commonly ordered pizza size.
5. Top Pizzas: Listing the top 5 most ordered pizza types along with their quantities.

Advanced Queries and Analysis

Further questions expand on joining multiple tables to uncover deeper insights:
Determining the total quantity sold for each pizza category (e.g., Veg, Non-Veg, etc.).
Analyzing patterns in customer preferences across various pizza types, sizes, and pricing tiers.

Objective

The goal of the project is to provide actionable insights into sales trends, popular items, and revenue drivers, enabling businesses to optimize their strategies and better understand their customers.

QUESTIONS SOLVED

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.

- Join the necessary tables to find the total quantity of each pizza category.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```sql
SELECT
    COUNT(ORDER_ID) AS TOTAL_ORDERS
FROM
    ORDERS;
```

TOTAL_ORDERS
21350

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 ON PIZZAS.PIZZA_ID = ORDER_DETAILS.PIZZA_ID;
```

Result Grid	
	TOTAL_SALES
▶	817860.05

IDENTIFY THE HIGHEST PRICE PIZZA

```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;
```

Result Grid	Filter Row
NAME	**PRICE**
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.


```sql
    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 Rows

SIZE	ORDER_COUNT
L	18526

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

```sql
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 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.

```sql
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 Row

CATEGORY	QUANTITY
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

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 Ro

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

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

```sql
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
SELECT
    ROUND(AVG(QUANTITY), 0) AS AVG_PIZZA_ORDERED_PER_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	Filter Rows:
AVG_PIZZA_ORDERED_PER_DAY	
▶ 138	

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:

NAME	REVENUE
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

THANK YOU