PIZZA SALES ANALYSIS

BY USING

SQL

Presented by

Inturi Suparna Babu

AGENDA

- About the Project
- List of tables used
- In detailed analysis
- Key take aways

ABOUT THE PROJECT

Hello Data Planet, my name is Inturi Suparna Babu, and I am excited to present my project on Pizza Sales Analysis using SQL. This project focuses on analysing pizza sales data for the year 2015. Through this analysis, I have explored various aspects of pizza sales, including the number of orders, revenue generation, and other insightful metrics.

By utilizing SQL, I have been able to extract meaningful insights from the dataset, which consists of detailed information on pizza types, prices, and order quantities. This analysis aims to provide a comprehensive understanding of the sales dynamics and performance of different pizza offerings over the course of the year.

Join me as I delve into the data and uncover key trends and patterns that can help drive business decisions and strategies for optimizing pizza sales.

LIST OF TABLES USED

In this analysis, I utilized four tables outlined below:

- ❑ Pizzas
- ❑ Pizza Types
- ❑ Pizza Orders
- ❑ Pizza Order Details

pizza_types
- Columns
  - pizza_type_id
  - name
  - category
  - ingredients

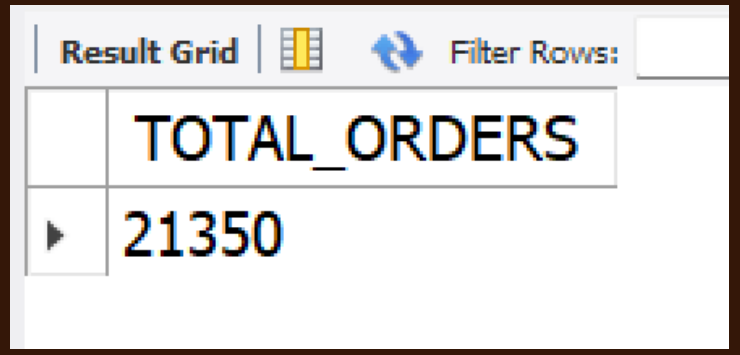
pizzas
- Columns
  - pizza_id
  - pizza_type_id
  - size

pizza_orders
- Columns
  - order_id
  - date
  - time

pizza_order_details
- Columns
  - order_details_id
  - order_id
  - pizza_id
  - quantity

Query The Total Number Of Orders Placed?

```
SELECT
    COUNT(ORDER_ID) AS TOTAL_ORDERS
FROM
    PIZZA_ORDERS;
```



Query The Total Number Of Pizza Id Types?

```sql
SELECT
    COUNT(DISTINCT PIZZA_ID) AS TOTAL_PIZZA_IDs
FROM
    PIZZAS;
```

Result Grid | Filter Rows:

TOTAL_PIZZA_IDs
96

Query The Total Number Of Pizza Types?

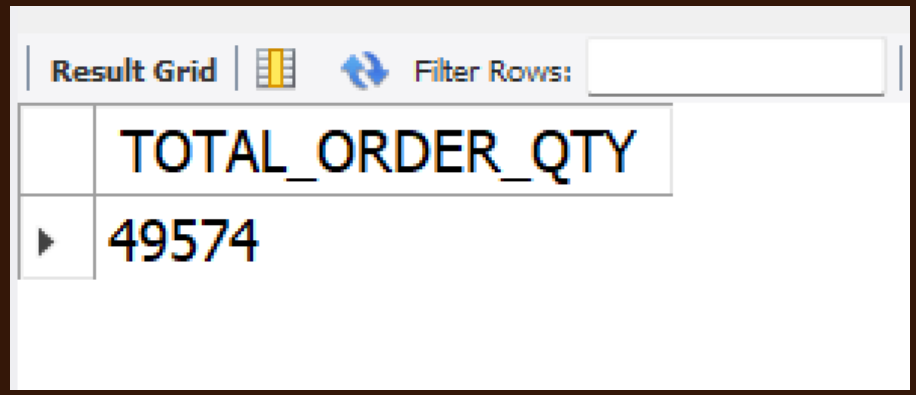
```sql
SELECT
    COUNT(DISTINCT PIZZA_TYPE_ID) AS TOTAL_PIZZA_TYPEs
FROM
    PIZZAS;
```

Result Grid | Filter Rows:

TOTAL_PIZZA_TYPEs
32

Query Total Order Quantity Placed?

```
SELECT
    SUM(QUANTITY) AS TOTAL_ORDER_QTY
FROM
    PIZZA_ORDER_DETAILS;
```



Query The First Date Of Transction And Last Date Of Transaction?

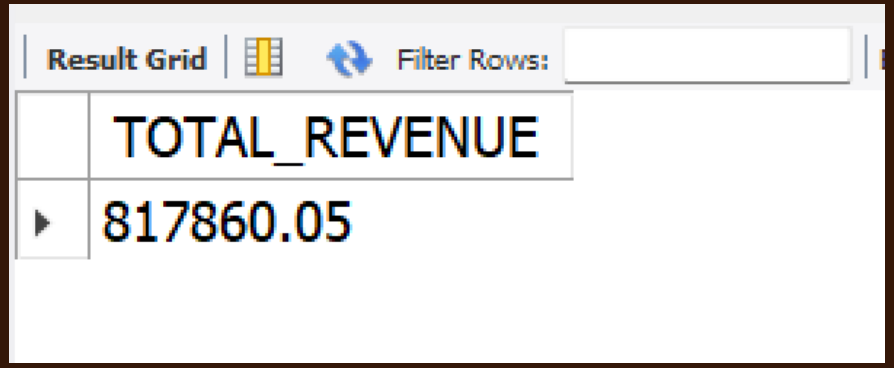
```sql
SELECT
    MIN(DATE) AS FIRST_DATE, MAX(DATE) AS LAST_DATE
FROM
    PIZZA_ORDERS;
```

Result Grid | Filter Rows:

	FIRST_DATE	LAST_DATE
▸	2015-01-01	2015-12-31

Query The Total Revenue Generated ?

```sql
SELECT
    ROUND(SUM(PRICE * QUANTITY), 2) AS TOTAL_REVENUE
FROM
    PIZZAS A
        JOIN
    PIZZA_ORDER_DETAILS B ON A.PIZZA_ID = B.PIZZA_ID;
```



Query The Highest Priced Pizza?
#Display Pizza Name, Price

```
SELECT
    A.NAME AS PIZZA_NAME, B.PRICE
FROM
    PIZZA_TYPES A
        JOIN
    PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID
ORDER BY B.PRICE DESC
LIMIT 1;
```

Result Grid | Filter Rows:

PIZZA_NAME	PRICE
The Greek Pizza	35.95

Query The Lowset Priced Pizza?
#Display Pizza Name, Price

```sql
SELECT
    A.NAME, B.PRICE
FROM
    PIZZA_TYPES A
        JOIN
    PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID
ORDER BY B.PRICE ASC
LIMIT 1;
```

Result Grid | Filter Rows:

NAME	PRICE
The Pepperoni Pizza	9.75

Query The Pizza Ordered Qty By Size?

```sql
SELECT
    A.SIZE, SUM(B.QUANTITY) AS ORDER_QTY
FROM
    PIZZAS A
        JOIN
    PIZZA_ORDER_DETAILS B ON A.PIZZA_ID = B.PIZZA_ID
GROUP BY A.SIZE
ORDER BY ORDER_QTY DESC;
```

Result Grid | Filter Rows:

SIZE	ORDER_QTY
L	18956
M	15635
S	14403
XL	552
XXL	28

Query The Pizza Ordered Qty By Category?

```sql
SELECT
    C.CATEGORY, SUM(D.QUANTITY) AS ORDER_QTY
FROM
    (SELECT
        A.CATEGORY, B.PIZZA_ID
    FROM
        PIZZA_TYPES A
    JOIN PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) C
        JOIN
    PIZZA_ORDER_DETAILS D ON C.PIZZA_ID = D.PIZZA_ID
GROUP BY C.CATEGORY
ORDER BY ORDER_QTY DESC;
```

Result Grid | Filter Rows:

CATEGORY	ORDER_QTY
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Query The Pizza Ordered Qty By Pizza Name?

```sql
SELECT
    C.NAME AS PIZZA_NAME, SUM(D.QUANTITY) AS ORDER_QTY
FROM
    (SELECT
        A.NAME, B.PIZZA_ID
    FROM
        PIZZA_TYPES A
    JOIN PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) C
        JOIN
    PIZZA_ORDER_DETAILS D ON C.PIZZA_ID = D.PIZZA_ID
GROUP BY PIZZA_NAME
ORDER BY ORDER_QTY DESC;
```

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

PIZZA_NAME	ORDER_QTY
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Query The Top 7 Types Based Orders Qty? Display Pizza Name, Order Qty

```sql
SELECT
    C.NAME, SUM(D.QUANTITY) AS TOTAL_ORDERQTY
FROM
    (SELECT
        A.NAME, B.PIZZA_ID
    FROM
        PIZZA_TYPES A
    JOIN PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) C
        JOIN
    PIZZA_ORDER_DETAILS D ON C.PIZZA_ID = D.PIZZA_ID
GROUP BY C.NAME
ORDER BY TOTAL_ORDERQTY DESC
LIMIT 7;
```

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

NAME	TOTAL_ORDERQTY
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371
The California Chicken Pizza	2370
The Sicilian Pizza	1938

Query The Distribution Of Orders By Hour Of Day?

```
SELECT
    HOUR(TIME) AS HOUR, COUNT(ORDER_ID) AS ORDERQTY
FROM
    PIZZA_ORDERS
GROUP BY HOUR
ORDER BY ORDERQTY DESC;
```

Result Grid | Filter Rows:

HOUR	ORDERQTY
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642

Query The Pizza Order Qty By Date And Calculate The Average Numbers Of Pizzas Ordered Per Day?

```sql
SELECT
    ROUND(AVG(ORDEREDQTY)) AS AVGORDERPIZZAQTY
FROM
    (SELECT
        A.DATE, SUM(B.QUANTITY) AS ORDEREDQTY
    FROM
        PIZZA_ORDERS A
    JOIN PIZZA_ORDER_DETAILS B ON A.ORDER_ID = B.ORDER_ID
    GROUP BY A.DATE
    ORDER BY ORDEREDQTY DESC) AS C;
```

Result Grid | Filter Rows:

AVGORDERPIZZAQTY
138

Query The Top 7 Pizza Names By Revenue?
#Display Pizza Name, Revenue , Orderqty

```sql
SELECT
    C.NAME,
    SUM(C.PRICE * E.QUANTITY) AS REVENUE,
    SUM(QUANTITY) AS TTLORDERQTY
FROM
    (SELECT
        A.NAME, B.PRICE, B.PIZZA_ID
    FROM
        PIZZA_TYPES A
    JOIN PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) C
        JOIN
    PIZZA_ORDER_DETAILS E ON C.PIZZA_ID = E.PIZZA_ID
GROUP BY C.NAME
ORDER BY REVENUE DESC
LIMIT 7;
```

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

NAME	REVENUE	TTLORDERQTY
The Thai Chicken Pizza	43434.25	2371
The Barbecue Chicken Pizza	42768	2432
The California Chicken Pizza	41409.5	2370
The Classic Deluxe Pizza	38180.5	2453
The Spicy Italian Pizza	34831.25	1924
The Southwest Chicken Pi...	34705.75	1917
The Italian Supreme Pizza	33476.75	1884

Query The Percentage Contribution Of Each Pizza Category To Total Revenue?

```sql
SELECT
    A.CATEGORY,
    ROUND(SUM(A.PRICE * C.QUANTITY) / (SELECT
                SUM(A.PRICE * B.QUANTITY) AS TOTAL_REVENUE
            FROM
                PIZZAS A
                    JOIN
                PIZZA_ORDER_DETAILS B ON A.PIZZA_ID = B.PIZZA_ID) * 100,
        2) AS REVENUEPERCENTAGE
FROM
    (SELECT
        A.CATEGORY, B.PIZZA_ID, B.PRICE
    FROM
        PIZZA_TYPES A
    JOIN PIZZAS B ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) A
        JOIN
    PIZZA_ORDER_DETAILS C ON A.PIZZA_ID = C.PIZZA_ID
GROUP BY A.CATEGORY
ORDER BY REVENUEPERCENTAGE DESC;
```

Result Grid | Filter Rows: | Export:

CATEGORY	REVENUEPERCENTAGE
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Analyse the cumulative revenue generated over time?

```sql
SELECT
  DATE,
  ROUND(SUM(TTLREVENUE) OVER(ORDER BY DATE),2) AS CUM_REVENUE
FROM
  (SELECT B.DATE, SUM(REVENUE) AS TTLREVENUE
    FROM (SELECT
          A.ORDER_ID,
          SUM(A.QUANTITY * B.PRICE) AS REVENUE
        FROM
          PIZZA_ORDER_DETAILS A
          JOIN PIZZAS B ON A.PIZZA_ID = B.PIZZA_ID
        GROUP BY
          A.ORDER_ID
      ) A
      JOIN PIZZA_ORDERS B ON A.ORDER_ID = B.ORDER_ID
    GROUP BY
      B.DATE
  ) AS RVN;
```

Result Grid | Filter Rows:

DATE	CUM_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

Determine The Top 3 Most Ordered Pizza Types Based On Revenue For Each Category?

```sql
SELECT NAME,CATEGORY,REVENUE FROM(
SELECT *, RANK() OVER(PARTITION BY CATEGORY ORDER BY REVENUE DESC) AS RNK
FROM(
SELECT A.NAME,A.CATEGORY,SUM(B.QUANTITY * A.PRICE) AS REVENUE
FROM (
SELECT A.NAME,A.CATEGORY,B.PIZZA_ID,B.PRICE
FROM PIZZA_TYPES A
JOIN PIZZAS B
ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID) A
JOIN PIZZA_ORDER_DETAILS B
ON A.PIZZA_ID = B.PIZZA_ID
GROUP BY A.NAME,A.CATEGORY) A) B
WHERE RNK <=3;
```

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

NAME	CATEGORY	REVENUE
The Thai Chicken Pizza	Chicken	43434.25
The Barbecue Chicken Pizza	Chicken	42768
The California Chicken Pizza	Chicken	41409.5
The Classic Deluxe Pizza	Classic	38180.5
The Hawaiian Pizza	Classic	32273.25
The Pepperoni Pizza	Classic	30161.75
The Spicy Italian Pizza	Supreme	34831.25
The Italian Supreme Pizza	Supreme	33476.75
The Sicilian Pizza	Supreme	30940.5

Key Takeaways

- ✓ The most ordered pizza size was Large, with an order quantity of 18,956, while the XXL size is underperforming.
- ✓ The Classic pizza category had the highest order quantity at 14,888, whereas the Chicken category had the lowest at 11,050 orders.
- ✓ Most orders occur during afternoon lunch hours (12:00-13:00) and evening hours (17:00-19:00).
- ✓ The average pizza order quantity is 138.
- ✓ The top 7 pizzas based on order quantity are:
  - The Classic Deluxe Pizza
  - The Barbecue Chicken Pizza
  - The Hawaiian Pizza
  - The Pepperoni Pizza
  - The Thai Chicken Pizza
  - The California Chicken Pizza
  - The Sicilian Pizza
- ✓ The Classic category generates the highest revenue percentage at 26.91%.

THANK YOU FOR WATCHING

Follow

Save

If you find this post helpful, I'd appreciate it if you could **like, share,** and **follow** me for more updates and insights

React

Share

Inturi Suparna Babu

Data Analyst

Microsoft Certified: Power BI Data Analyst Associate (PL 300)