This page is the SQL queries report for cross-validation analytical results purposes.

The Analysis Project Report and Recommendations:

https://github.com/NguyenDangXuanLinh/The-Pizza-Analysis-Report-with-PowerBI-SQL

Power BI Dashboard:

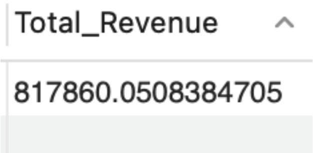
https://app.powerbi.com/view?r=eyJrIjoiMDMwMjRjMjQtYzJlNC00OTgwLWIxYmEtNDRkMjZkNjg3NDI0IiwidCI6ImRmODY3OWNkLWE4MGUtNDVkOC05OWFjLWM4M2VkN2ZmOTVhMCJ9

PIZZA SALES SQL QUERIES

A. KPI'S

1. Total Revenue:

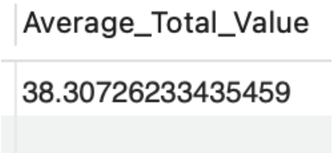
```
SELECT SUM(total_price) AS Total_Revenue
FROM pizza_sales;
```



Total_Revenue
817860.0508384705

2. Average Order Value:

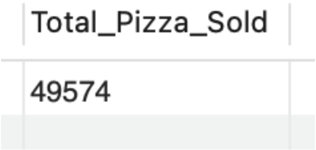
```
SELECT SUM(total_price) / COUNT(DISTINCT (order_id)) AS Average_Total_Value
FROM pizza_sales;
```



Average_Total_Value
38.30726233435459

3. Total Pizza Sold:

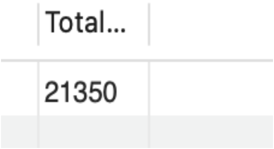
```
SELECT SUM(quantity) AS Total_Pizza_Sold
FROM pizza_sales;
```



Total_Pizza_Sold
49574

4. Total Orders:

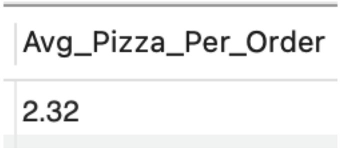
```
SELECT COUNT(DISTINCT (order_id)) AS Total_Orders
FROM pizza_sales;
```



Total...
21350

5. Average Pizza Per Order:

```
SELECT
CAST(CAST(SUM(quantity) AS DECIMAL(10,2)) /
CAST(COUNT(DISTINCT(order_id)) AS DECIMAL(10,2)) AS DECIMAL(10,2)) AS
Avg_Pizza_Per_Order

FROM pizza_sales;
```



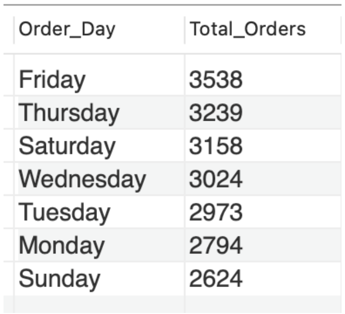
Avg_Pizza_Per_Order
2.32

B. Daily Trends for Total Orders:

```
--Convert order_date column's data type to Datetime--
UPDATE pizza_sales
  SET order_date = STR_TO_DATE(order_date, "%d/%m/%Y");
--------------------------------------------------------------------------------
```

```
SELECT DATE_FORMAT(order_date, "%W") AS Order_Day,
               COUNT(DISTINCT order_id) AS Total_Orders
  FROM pizza_sales
GROUP BY DATE_FORMAT(order_date, "%W")
ORDER Total_Orders DESC;
```

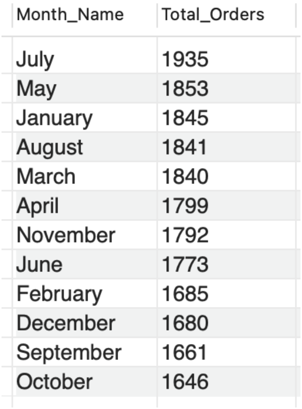
Output:



Order_Day	Total_Orders
Friday	3538
Thursday	3239
Saturday	3158
Wednesday	3024
Tuesday	2973
Monday	2794
Sunday	2624

C. Hourly Trend for Total Orders:

```
SELECT DATE_FORMAT (order_date, "%M") AS Month_Name,
               COUNT(DISTINCT order_id) AS Total_Orders
  FROM pizza_sales
GROUP BY DATE_FORMAT (order_date, "%M")
ORDER BY Total_Orders DESC;
```

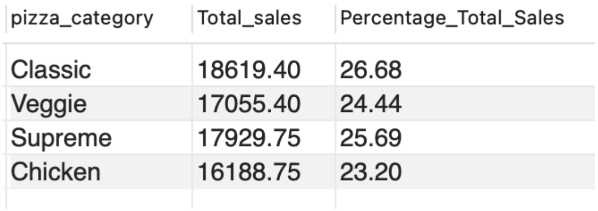


Month_Name	Total_Orders
July	1935
May	1853
January	1845
August	1841
March	1840
April	1799
November	1792
June	1773
February	1685
December	1680
September	1661
October	1646

D. Percentage of Sales by Pizza Category:

```
SELECT  pizza_category,
                CAST(SUM(total_price) AS DECIMAL( 10,2) ) AS Total_sales,
                CAST(SUM(total_price) * 100 / (SELECT SUM(total_price)
                                              FROM pizza_sales
                                            WHERE MONTH(order_date) = 1) AS DECIMAL (10,2))
                                        AS Percentage_Total_Sales


FROM pizza_sales
WHERE MONTH(order_date) = 1
GROUP BY pizza_category;
```

Output

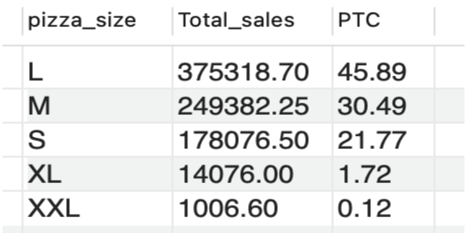


pizza_category	Total_sales	Percentage_Total_Sales
Classic	18619.40	26.68
Veggie	17055.40	24.44
Supreme	17929.75	25.69
Chicken	16188.75	23.20

E. Percentage of Sales by Pizza Size:

```
SELECT          pizza_size,
                CAST(SUM(total_price) AS DECIMAL( 10,2) ) AS Total_sales,
                CAST(SUM(total_price) * 100 / (SELECT SUM(total_price)
                                                      FROM pizza_sales ) AS DECIMAL(10,2)) AS
PTC

FROM pizza_sales
GROUP BY pizza_size
ORDER BY PTC DESC;
```

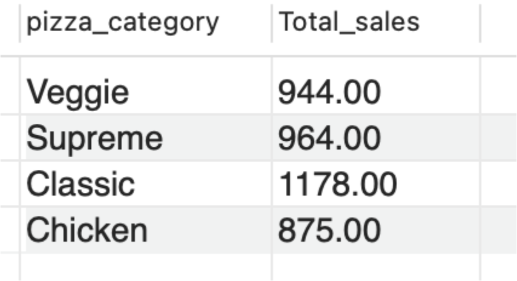
Output



pizza_size	Total_sales	PTC
L	375318.70	45.89
M	249382.25	30.49
S	178076.50	21.77
XL	14076.00	1.72
XXL	1006.60	0.12

F. Total of Pizzas Sold by Category (per month):

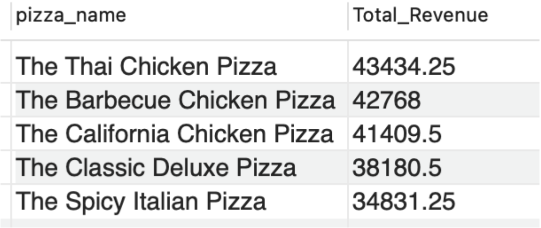
```
SELECT          pizza_category,
                CAST(SUM(quantity) AS DECIMAL(10,2)) AS Total_sales

FROM pizza_sales
WHERE MONTH(order_date) = 2
GROUP BY pizza_category;
```



pizza_category	Total_sales
Veggie	944.00
Supreme	964.00
Classic	1178.00
Chicken	875.00

G. Top 5 Pizzas By Revenue:

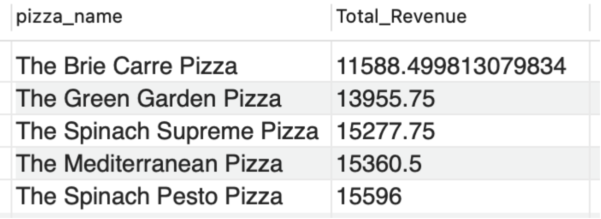
```
SELECT  pizza_name,
                SUM(total_price) AS Total_Revenue
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue DESC
LIMIT 5;
```



pizza_name	Total_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 Spicy Italian Pizza	34831.25

H. Bottom 5 Pizzas By Revenue:

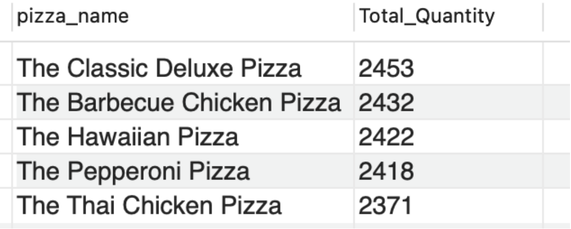
```
SELECT  pizza_name,
                    SUM(total_price) AS Total_Revenue
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue
LIMIT 5;
```



pizza_name	Total_Revenue
The Brie Carre Pizza	11588.499813079834
The Green Garden Pizza	13955.75
The Spinach Supreme Pizza	15277.75
The Mediterranean Pizza	15360.5
The Spinach Pesto Pizza	15596

I. Top 5 Pizzas By Quantity:

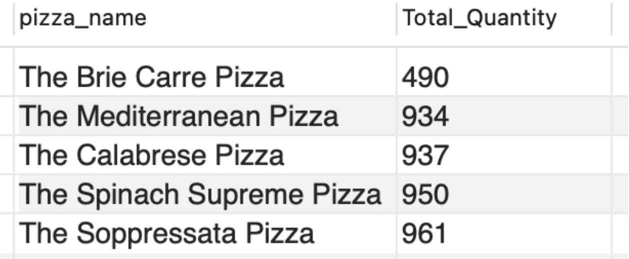
```
SELECT  pizza_name,
                    SUM(quantity) AS Total_Quantity
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Quantity DESC
LIMIT 5;
```



pizza_name	Total_Quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

J. Bottom 5 Pizzas By Quantity:

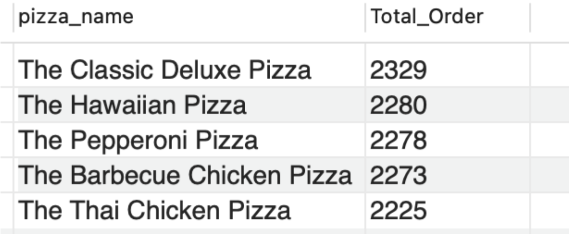
```
SELECT  pizza_name,
                    SUM(quantity) AS Total_Quantity
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Quantity
LIMIT 5;
```



pizza_name	Total_Quantity
The Brie Carre Pizza	490
The Mediterranean Pizza	934
The Calabrese Pizza	937
The Spinach Supreme Pizza	950
The Soppressata Pizza	961

K. Top 5 Pizzas By Order:

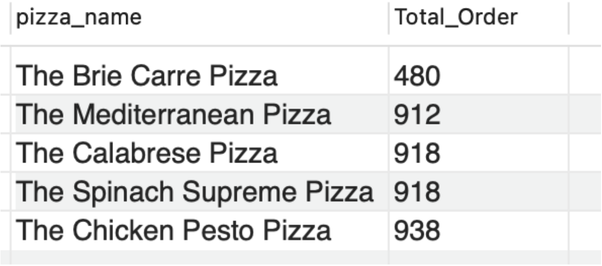
```
SELECT  pizza_name,
                  COUNT(DISTINCT order_id) AS Total_Order
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Order DESC
LIMIT 5;
```



pizza_name	Total_Order
The Classic Deluxe Pizza	2329
The Hawaiian Pizza	2280
The Pepperoni Pizza	2278
The Barbecue Chicken Pizza	2273
The Thai Chicken Pizza	2225

L. Bottom 5 Pizzas By Order:

```
SELECT  pizza_name,
                  COUNT(DISTINCT order_id) AS Total_Order
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Order
LIMIT 5;
```



pizza_name	Total_Order
The Brie Carre Pizza	480
The Mediterranean Pizza	912
The Calabrese Pizza	918
The Spinach Supreme Pizza	918
The Chicken Pesto Pizza	938

NOTE

If you want to apply the Month, Quarter, Week filters to the queries you can use WHERE clause. Follow some of below examples:

```
SELECT pizza_category, SUM(total_price) AS Total_sales,
         SUM(total_price) * 100 / (SELECT SUM(total_price)
                                   FROM pizza_sales
                                 WHERE MONTH(order_date) = 1
                                   ) AS Percentage_Total_Sale
  FROM pizza_sales
WHERE MONTH(order_date) = 1
GROUP BY pizza_category;
```

** Here MONTH(order_date) = 1 indicates that the output is for the month of January.*

** MONTH(order_date) = 4 is output for April*

```
SELECT pizza_category, SUM(total_price) AS Total_sales,
         SUM(total_price) * 100 / (SELECT SUM(total_price)
                                   FROM pizza_sales
                                 WHERE QUARTER (order_date) = 1
                                   ) AS Percentage_Total_Sales
 FROM pizza_sales
WHERE QUARTER (order_date) = 1
GROUP BY pizza_category;
```

** Here QUARTER(order_date) = 1 indicates that the output is for the Quarter 1.*

** QUARTER(order_date) = 4 is output for Quarter 4.*