Pizza Sales Portfolio Project – SQL & Power BI

This project comes in 4 CSV files. In this project, I’ve used Microsoft SQL server 2019 to uncover valuable insights to help the pizza outlet make important decisions to increase sales.

The data is presented using Power BI and includes KPIs and other questions important to uncovering these insights.

The raw data for this project is presented in 4 CSV files. This data represents a year’s worth of sales for a pizza outlet, and they need to answer a few questions which will help them make important decisions to increase sales and improve their business.

The project is done in Microsoft SQL server and presented in Power BI. The data was loaded into 4 tables. This project involves the use of simple joins and sub-queries.

The project seeks to answer the following questions:

-- KPIs

-- 1) Total Revenue (How much money did we make this year?)

-- 2) Average Order Value

-- 3) Total Pizzas Sold

-- 4) Total Orders

-- 5) Average Pizzas per Order

-- QUESTIONS TO ANSWER

-- 1) Daily Trends for Total Orders

-- 2) Hourly Trend for Total Orders

-- 3) Percentage of Sales by Pizza Category

-- 4) Percentage of Sales by Pizza Size

-- 5) Total Pizzas Sold by Pizza Category

-- 6) Top 5 Best Sellers by Total Pizzas Sold

-- 7) Bottom 5 Worst Sellers by Total Pizzas Sold

FINDINGS:

-- KPIs

-- 1) Total Revenue for the year was $817,860

-- 2) Average Order Value was $38.31

-- 3) Total Pizzas Sold – 50,000

-- 4) Total Orders – 21,000

-- 5) Average Pizzas per Order – 2

-- QUESTIONS to Answerd

-- 1) The busiest days are Thursday (3239 orders), Friday (3538 orders) and Saturday (3158 orders). Most sales are recorded on Friday

-- 2) Most orders are placed between 12pm to 1pm, and 5pm to 7pm

-- 3) Classic pizza has the highest percentage sales (26.91%), followed by Supreme (25.46%), Chicken (23.96%) and Veggie (23.68%) pizzas

-- 4) Large size pizzas record the highest sales (45.89%) followed by medium (30.49%), then small (21.77%). XL and XXL only account for 1.72% and 0.12% respectively

-- 5) Classic Pizza accounts for the highest sales (14,888 pizzas) followed by Supreme (11,987 pizzas), Veggie (11,649 pizzas) and Chicken (11,050 pizzas)

-- 6) Top 5 Best Sellers are the Classic Deluxe (2453 pizzas), Barbecue Chicken (2432 pizzas), Hawaiian (2422), Peperoni (2418 pizzas) and Thai Chicken (2371 pizzas)

-- 7) Bottom 5 Worst Sellers are Brie Carre (490 pizzas), Mediterranean (934 pizzas), Calabrese (937 pizzas), Spinach Supreme (950 pizzas) and Soppressata (961).

CONCLUSION:

The outlet should capitalize on Large size Classic, Supreme, Veggie and Chicken pizzas.

Since XL and XXL pizzas account for such a small percentage of their sales (just 1.94%), they can safely get rid of these pizza sizes.

Even though the Brie Carre pizza is the worst seller, it recorded 490 pizzas sold. It would still be a good idea to keep it in the menu.

QUERIES USED:

USE pizza;

SELECT \*

from pizzas

-- KPIs

-- 1) Total Revenue

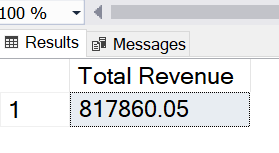
SELECT

round(SUM(quantity \* price), 2)

FROM order\_details AS o

JOIN pizzas AS p

ON o.pizza\_id = p.pizza\_id



-- 2) Average Order Value

-- total order value/order count

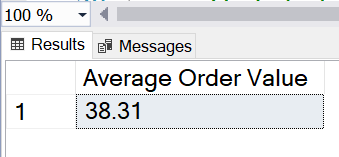
SELECT

SUM(quantity \* price)/ COUNT(DISTINCT order\_id) AS [Average Order Value]

FROM order\_details AS o

JOIN pizzas AS p

ON o.pizza\_id = p.pizza\_id



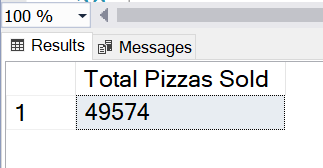
-- 3) Total pizzas sold

SELECT

SUM(quantity) AS [Total Pizzas Sold]

FROM

order\_details



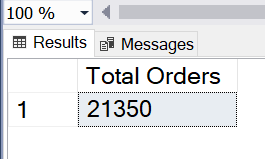
-- 4) Total Orders

SELECT

COUNT(DISTINCT order\_id) AS [Total Orders]

FROM

order\_details



-- 5) Average Pizzas Per Order

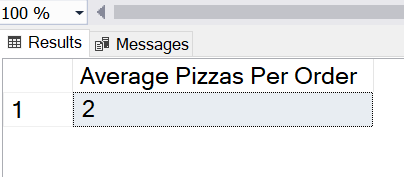
-- quantity sold/order IDs

SELECT

ROUND(SUM(quantity)/COUNT(DISTINCT order\_id),2) AS [Average Pizzas Per Order]

FROM

order\_details



-- Sales analysis

-- 1) Daily Trends for Total Orders

SELECT

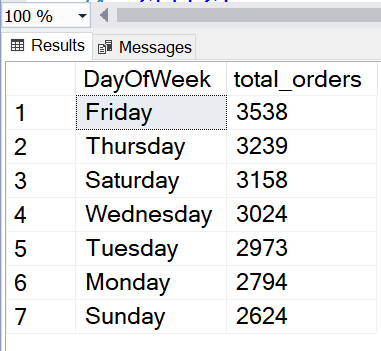
FORMAT(date, 'dddd') AS DayOfWeek

,COUNT(DISTINCT order\_id) AS total\_orders

FROM orders

GROUP BY FORMAT(date, 'dddd')

ORDER BY total\_orders DESC



-- 2) Hourly TrendS for Total Orders

SELECT

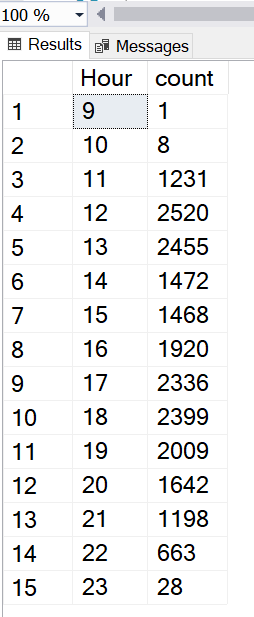
DATEPART(HOUR, time) AS [Hour]

,COUNT(DISTINCT order\_id) AS Total\_Orders

FROM orders

GROUP BY DATEPART(HOUR, time)

ORDER BY [Hour]



-- 3) Percentage of Sales by Pizza Category

SELECT \*

FROM

pizzas

SELECT

category,

ROUND(SUM(quantity \* price), 2) AS revenue,

ROUND(SUM(quantity \* price) \* 100.0 / (SELECT SUM(quantity \* price) FROM pizzas AS p2 JOIN order\_details AS od2 ON od2.pizza\_id = p2.pizza\_id), 2) AS percentage\_of\_sales

FROM

pizzas AS p

JOIN

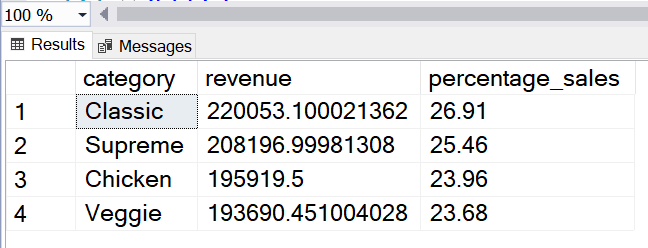
pizza\_types AS pt ON p.pizza\_type\_id = pt.pizza\_type\_id

JOIN

order\_details AS od ON od.pizza\_id = p.pizza\_id

GROUP BY

category;



-- 4) Percentage of Sales by Pizza Size

SELECT

size

,ROUND(SUM(quantity \* price), 2) AS revenue

,ROUND(SUM(quantity \* price) \* 100.0 / (SELECT SUM(quantity \* price) FROM pizzas AS p2 JOIN order\_details AS od2 ON od2.pizza\_id = p2.pizza\_id), 2) AS percentage\_of\_sales

FROM

pizzas AS p

JOIN

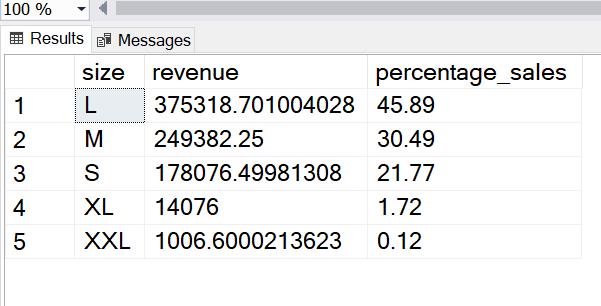
pizza\_types AS pt ON p.pizza\_type\_id = pt.pizza\_type\_id

JOIN

order\_details AS od ON od.pizza\_id = p.pizza\_id

GROUP BY

size;



-- 5) Total Pizzas Sold by Pizza Category

SELECT

category

,SUM(quantity) AS quantity\_sold

FROM

pizzas AS p

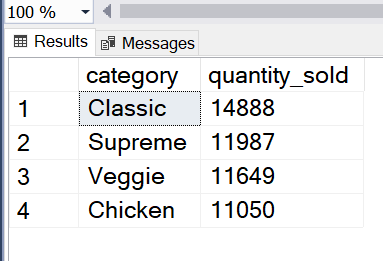
JOIN

pizza\_types AS pt ON p.pizza\_type\_id = pt.pizza\_type\_id

JOIN

order\_details AS od ON od.pizza\_id = p.pizza\_id

GROUP BY category;



-- 6) Top 5 Best Sellers by Total Pizzas Sold

SELECT top 5

name

,SUM(quantity) AS total\_quantity\_sold

FROM

pizzas AS p

JOIN

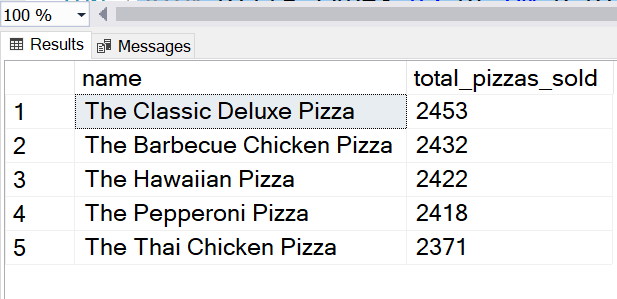
pizza\_types AS pt ON p.pizza\_type\_id = pt.pizza\_type\_id

JOIN

order\_details AS od ON od.pizza\_id = p.pizza\_id

GROUP BY name

ORDER BY total\_quantity\_sold DESC;



-- 7) Bottom 5 Best Sellers by Total Pizzas Sold

SELECT top 5

name

,SUM(quantity) AS total\_quantity\_sold

FROM

pizzas AS p

JOIN

pizza\_types AS pt ON p.pizza\_type\_id = pt.pizza\_type\_id

JOIN

order\_details AS od ON od.pizza\_id = p.pizza\_id

GROUP BY name

ORDER BY total\_quantity\_sold ASC;

