**PIZZA SALES ANALYSIS**

-- Dropping the schema "pizza\_sales\_schema" if already exists, to get rid of error.

DROP SCHEMA IF EXISTS PIZZA\_SALES\_SCHEMA;

-- Creating a Schema named "pizza\_sales\_schema"

CREATE SCHEMA PIZZA\_SALES\_SCHEMA;

-- Making use of pizza\_sales\_schema as the default schema for the further queries.

USE PIZZA\_SALES\_SCHEMA;

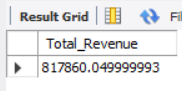
-- Viewing the data in the pizza\_sales table.

SELECT \* FROM pizza\_sales;

**KPI Requirements:**

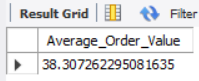
1. **Total Revenue:** Sum of Total Price of all pizza orders.

SELECT SUM(total\_price) AS Total\_Revenue FROM pizza\_sales;

****

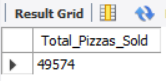
1. **Average Order Value:** The average amount spent per order, calculated by dividing total revenue by total number of orders.

SELECT (SUM(total\_price) / COUNT(DISTINCT(order\_id))) AS Average\_Order\_Value FROM pizza\_sales;



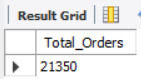
1. **Total Pizzas Sold:** Sum of the quantities of all pizzas sold.

SELECT SUM(quantity) AS Total\_Pizzas\_Sold FROM pizza\_sales;



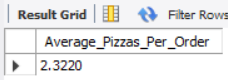
1. **Total Orders:** Total number of orders placed.

SELECT COUNT(DISTINCT(order\_id)) AS Total\_Orders FROM pizza\_sales;



1. **Average Pizzas Per Order:** Average number of pizzas sold per order, calculated by dividing total\_pizzas\_sold by total number of orders.

SELECT (SUM(quantity) / COUNT(DISTINCT(order\_id))) AS Average\_Pizzas\_Per\_Order FROM pizza\_sales;



**CHART Requirements:**

-- 1) Daily Trend for Total Orders

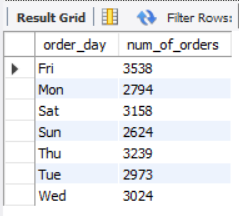
SELECT

DATE\_FORMAT(order\_date, '%a') AS order\_day,

COUNT(DISTINCT order\_id) AS num\_of\_orders

FROM pizza\_sales

GROUP BY 1;



-- 2) Monthly Trends for Total Orders

SELECT

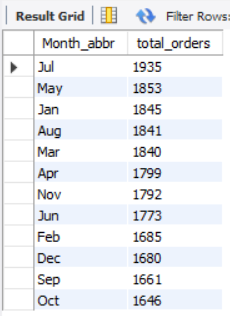
DATE\_FORMAT(order\_date, '%b') AS Month\_abbr,

COUNT(DISTINCT order\_id) AS total\_orders

FROM pizza\_sales

GROUP BY 1

ORDER BY 2 DESC;



-- 3) Percentage of Sales by Pizza Category

SELECT

Pizza\_Category,

ROUND(SUM(total\_price), 2) AS Total\_Sales,

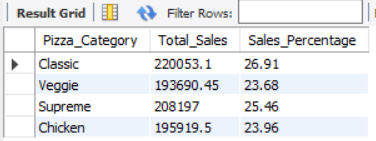
ROUND((SUM(total\_price) / (SELECT SUM(total\_price) FROM pizza\_sales)) \* 100, 2) AS Sales\_Percentage

FROM

pizza\_sales

GROUP BY

Pizza\_Category;



-- 4) Percentage of Sales by Pizza Size

SELECT

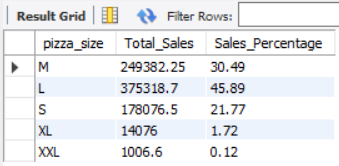
pizza\_size,

ROUND(SUM(total\_price), 2) AS Total\_Sales,

ROUND((SUM(total\_price) / (SELECT SUM(total\_price) FROM pizza\_sales)) \* 100, 2) AS Sales\_Percentage

FROM pizza\_sales

GROUP BY 1;



-- 5) Total Pizzas sold by Pizza Category

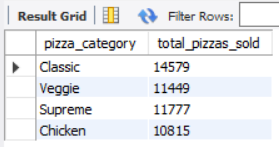
SELECT

pizza\_category,

COUNT(pizza\_id) AS total\_pizzas\_sold

FROM pizza\_sales

GROUP BY 1;



-- 6) Top 5 Best Sellers based on revenue

SELECT

pizza\_name,

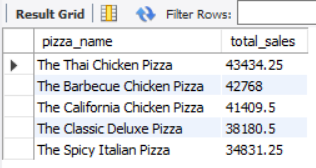
SUM(total\_price) AS total\_sales

FROM pizza\_sales

GROUP BY 1

ORDER BY 2 DESC

LIMIT 5;



-- 7) Bottom 5 Worst Sellers based on revenue

SELECT

pizza\_name,

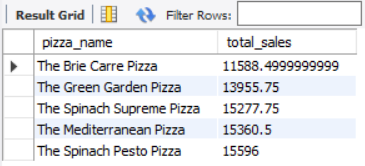
SUM(total\_price) AS total\_sales

FROM pizza\_sales

GROUP BY 1

ORDER BY 2

LIMIT 5;



-- 8) Top 5 Best Sellers based on Total Quantity

SELECT

pizza\_name,

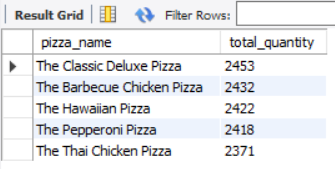
SUM(quantity) AS total\_quantity

FROM pizza\_sales

GROUP BY 1

ORDER BY 2 DESC

LIMIT 5;



-- 9) Bottom 5 Worst Sellers based on Total Quantity

SELECT

pizza\_name,

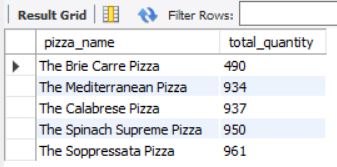
SUM(quantity) AS total\_quantity

FROM pizza\_sales

GROUP BY 1

ORDER BY 2

LIMIT 5;



-- 10) Top 5 Best Sellers based on Total Orders

SELECT

pizza\_name,

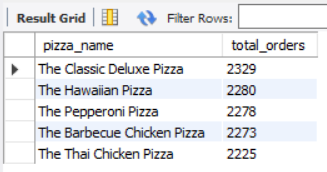
COUNT(DISTINCT order\_id) AS total\_orders

FROM pizza\_sales

GROUP BY 1

ORDER BY 2 DESC

LIMIT 5;



-- 11) Bottom 5 Worst Sellers based on Total Orders

SELECT

pizza\_name,

COUNT(DISTINCT order\_id) AS total\_orders

FROM pizza\_sales

GROUP BY 1

ORDER BY 2

LIMIT 5;

