**PIZZA SALES SQL QUERIES**

**KPI’s REQUIREMENT**

**1.Total Revenue: The sum of the total price of all pizza orders.**

**Total Revenue:**

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



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

**Average Order Value**

SELECT (SUM(total\_price) / COUNT(DISTINCT order\_id)) AS Avg\_order\_Value FROM pizza\_sales



**3.Total Pizzas Sold: The sum of the quantities of all pizzas sold.**

**Total Pizzas Sold**

SELECT SUM(quantity) AS Total\_pizza\_sold FROM pizza\_sales



**4. Total Orders: The total number of orders placed.**

**Total Orders**

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



**5.Average Pizzas Per Order: The average number of pizzas sold per order, calculated by dividing the total number of pizzas sold by the total number of orders.**

**Average Pizzas 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\_Pizzas\_per\_order

FROM pizza\_sales



**B. Hourly Trend for Total Pizzas Sold**

1. **Hourly Trend for Total Pizzas Sold:**

**Create a stacked bar chart that displays the hourly trend of total orders over a specific time period. This chart will help us identify any patterns or fluctuations in order volumes on a hourly basis.**

SELECT DATEPART(HOUR, order\_time) as order\_hours, SUM(quantity) as total\_pizzas\_sold

from pizza\_sales

group by DATEPART(HOUR, order\_time)

order by DATEPART(HOUR, order\_time)

***Output***

******

**Weekly Trend for Total Orders:**

**Create a line chart that illustrates the weekly trend of total orders throughout the year. This chart will allow us to identify peak weeks or periods of high order activity.**

**C. Weekly Trend for Orders**

SELECT

DATEPART(ISO\_WEEK, order\_date) AS WeekNumber,

YEAR(order\_date) AS Year,

COUNT(DISTINCT order\_id) AS Total\_orders

FROM

pizza\_sales

GROUP BY

DATEPART(ISO\_WEEK, order\_date),

YEAR(order\_date)

ORDER BY

Year, WeekNumber;

** **

**.Percentage of Sales by Pizza Category:**

**Create a pie chart that shows the distribution of sales across different pizza categories. This chart will provide insights into the popularity of various pizza categories and their contribution to overall sales**

**D. % of Sales by Pizza Category**

SELECT pizza\_category, CAST(SUM(total\_price) AS DECIMAL(10,2)) as total\_revenue,

CAST(SUM(total\_price) \* 100 / (SELECT SUM(total\_price) from pizza\_sales) AS DECIMAL(10,2)) AS PCT

FROM pizza\_sales

GROUP BY pizza\_category

***Output***

****

**4.Percentage of Sales by Pizza Size:**

**Generate a pie chart that represents the percentage of sales attributed to different pizza sizes. This chart will help us understand customer preferences for pizza sizes and their impact on sales.**

**E. % of Sales by Pizza Size**

SELECT pizza\_size, CAST(SUM(total\_price) AS DECIMAL(10,2)) as total\_revenue,

CAST(SUM(total\_price) \* 100 / (SELECT SUM(total\_price) from pizza\_sales) AS DECIMAL(10,2)) AS PCT

FROM pizza\_sales

GROUP BY pizza\_size

ORDER BY pizza\_size

***Output***

****

**Total Pizzas Sold by Pizza Category:**

**Create a funnel chart that presents the total number of pizzas sold for each pizza category. This chart will allow us to compare the sales performance of different pizza categories.**

**F. Total Pizzas Sold by Pizza Category**

SELECT pizza\_category, SUM(quantity) as Total\_Quantity\_Sold

FROM pizza\_sales

WHERE MONTH(order\_date) = 2

GROUP BY pizza\_category

ORDER BY Total\_Quantity\_Sold DESC

***Output***

****

**Top 5 Best & Worst Sellers by Revenue, Total Quantity and Total Orders**

**Create a bar chart highlighting the top 5 best & Worst-selling pizzas based on the Revenue, Total Quantity, Total Orders. This chart will help us identify the most popular pizza options.**

**G. Top 5 Pizzas by Revenue**

SELECT Top 5 pizza\_name, SUM(total\_price) AS Total\_Revenue

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Revenue DESC

****

**H. Bottom 5 Pizzas by Revenue**

SELECT Top 5 pizza\_name, SUM(total\_price) AS Total\_Revenue

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Revenue ASC

****

**I. Top 5 Pizzas by Quantity**

SELECT Top 5 pizza\_name, SUM(quantity) AS Total\_Pizza\_Sold

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Pizza\_Sold DESC

***Output***

****

**J. Bottom 5 Pizzas by Quantity**

SELECT TOP 5 pizza\_name, SUM(quantity) AS Total\_Pizza\_Sold

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Pizza\_Sold ASC

***Output***

****

**K. Top 5 Pizzas by Total Orders**

SELECT Top 5 pizza\_name, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Orders DESC

****

**L. Bottom 5 Pizzas by Total Orders**

SELECT Top 5 pizza\_name, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Orders ASC

******

***NOTE***

If you want to apply the pizza\_category or pizza\_size filters to the above queries you can use WHERE clause. Follow some of below examples

SELECT Top 5 pizza\_name, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

WHERE pizza\_category = 'Classic'

GROUP BY pizza\_name

ORDER BY Total\_Orders ASC