**PIZZA SALES SQL QUERIES**

**A.KPI’s**

**1. Total Revenue**

The sum of the total price of all pizza orders.

SELECT SUM(total\_price) AS Total\_Revenue

FROM pizza\_sales

**A screenshot of a message

Description automatically generated**

**2. Average Order Value**

The average amount spent per order, calculated by dividing the total revenue by the total number of orders.

SELECT (SUM(total\_price) / COUNT(DISTINCT order\_id)) AS Avg\_Order\_Value

FROM pizza\_sales

A screenshot of a computer message

Description automatically generated

**3.Total Pizza Sold**

The sum of the quantities of all pizzas sold.

SELECT SUM(quantity) AS Total\_Pizza\_Sold

A screenshot of a computer

Description automatically generatedFROM pizza\_sales

**4. Total Orders**

The total number of orders placed.

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

FROM pizza\_sales

A screenshot of a computer

Description automatically generated

**5. Average Pizza 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.

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

A screenshot of a message

Description automatically generated

**B. FOR VISUALIZATION**

**1. Daily Trend for Total Orders**

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

SELECT DATENAME(DW, order\_date) AS Order\_Day, COUNT(DISTINCT order\_id)

AS Total\_Orders

FROM pizza\_sales

A screenshot of a computer

Description automatically generatedGROUP BY DATENAME(DW, order\_date)

**2. Hourly Trend for Total Orders**

Create a line chart that illustrates the hourly trend of total orders throughout the day. This chart will allow us to identify peak hours or periods of high order activity.

SELECT DATEPART(HOUR, order\_time) as Order\_Hours, COUNT(DISTINCT order\_id)

AS Total\_Orders

FROM pizza\_sales

GROUP BY DATEPART(HOUR, order\_time)

A screenshot of a table

Description automatically generatedORDER BY DATEPART(HOUR, order\_time)

**3. 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.

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

A screenshot of a computer screen

Description automatically generated

**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.

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

A screenshot of a computer screen

Description automatically generatedORDER BY pizza\_size

**5. Total Pizza 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.

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

FROM pizza\_sales

GROUP BY pizza\_category

ORDER BY Total\_Quantity\_Sold DESC

A screenshot of a computer

Description automatically generated

**6. Top 5 Best Sellers by Total Pizza Sold**

Create a bar chart highlighting the top 5 best-selling pizzas based on the total number of pizzas sold.

This chart will help us identify the most popular pizza options.

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

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Pizza\_Sold DESC

A screenshot of a menu

Description automatically generated

**7. Top 5 Least Sellers by Total Pizza Sold**

Create a bar chart showcasing the bottom 5 worst-selling pizzas based on the total number of pizzas sold. This chart will enable us to identify underperforming or less popular pizza options.

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

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY Total\_Pizza\_Sold ASC

A screenshot of a menu

Description automatically generated

***NOTE!***

If you want to apply the Month, Quarter or Week filter to the above queries, you can use **WHERE** clause.

Follow some of below examples:

SELECT DATENAME(DW, order\_date) AS Order\_Day, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

WHERE MONTH(order\_date) = 1

GROUP BY DATENAME(DW, order\_date)

*\*Here, MONTH(order\_date) = 1 -indicates that the output is for the month of January.*

*MONTH(order\_date) = 4 -indicates output for month of April.*

SELECT DATENAME(DW, order\_date) AS Order\_Day, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

WHERE DATEPART(QUARTER, order\_date) = 1

GROUP BY DATENAME(DW, order\_date)

*\*Here, DATEPART(QUARTER, order\_date) = 1 -indicates that the output is for the 1st Quarter.*

*QUARTER(order\_date) = 3 -indicates output for 3rd Quarter.*

SELECT DATENAME(DW, order\_date) AS Order\_Day, COUNT(DISTINCT order\_id) AS Total\_Orders

FROM pizza\_sales

WHERE DATEPART( WEEK, order\_date) = 1

GROUP BY DATENAME(DW, order\_date)

*\*Here, DATEPART(WEEK, order\_date) = 1 -indicates that the output is for the 1st Week.*

*WEEK(order\_date) = 3 -indicates output for 3rd Week.*