**PIZZA SALES**

**KPI Requirements**

1. Total Revenue

select round(SUM(total\_price), 0) as total\_revenue

from pizza\_sales;



This query sums up all total\_price values in pizza\_sales to calculate total revenue, then rounds it to the nearest whole number.

2. Average Order Value

select round((SUM(total\_price) / count(distinct order\_id)), 2) as avg\_order\_value

from pizza\_sales;



Calculates the **average revenue per order** by dividing total sales (SUM(total\_price)) by the number of unique orders (count(distinct order\_id)), rounded to 2 decimals.

3. Total Pizzas Sold

select SUM(quantity) as total\_pizzas\_sold

from pizza\_sales;



Adds up all quantity values to get the **total number of pizzas sold**.

4. Total Orders

select COUNT(distinct order\_id) as total\_orders

from pizza\_sales;



Counts the number of unique order\_id values to find the **total orders placed**.

5. 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;



Divides total pizzas sold (SUM(quantity)) by total unique orders (COUNT(distinct order\_id)) to get the **average pizzas per order**, cast to 2 decimal places.

**TRENDS**

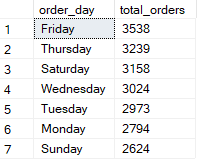
1. Daily Trends

select DATENAME(DW, order\_date) as order\_day, COUNT(distinct order\_id) as total\_orders

from pizza\_sales

group by DATENAME(DW, order\_date)

order by total\_orders desc;



Groups orders by the **day of the week** (DATENAME(DW, order\_date)) and counts unique order\_id values to show **total orders per weekday**.

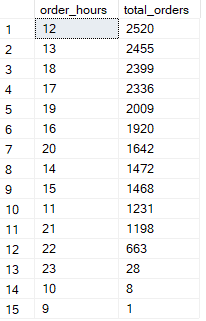
2. Hourly Trends

select DATEPART(HOUR, order\_time) as order\_hours, COUNT(distinct order\_id) as total\_orders

from pizza\_sales

group by DATEPART(HOUR, order\_time)

order by total\_orders desc;



Counts unique orders grouped by **hour of the day**, showing when most orders were placed.

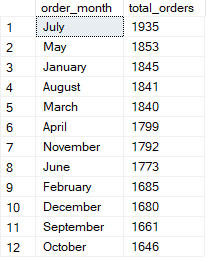
3. Monthly Trends

select DATENAME(MONTH, order\_date), COUNT(distinct order\_id) as total\_orders

from pizza\_sales

group by DATENAME(MONTH, order\_date)

order by total\_orders desc;



Groups orders by **month name** and counts unique order\_id values to get the **total orders per month**.

4. % of Sales by Pizza Category

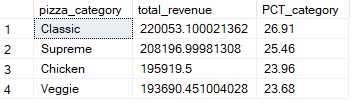
select pizza\_category, SUM(total\_price) as total\_revenue,

cast((SUM(total\_price) \* 100) / (select sum(total\_price) from pizza\_sales) as decimal(10,2)) as PCT\_category

from pizza\_sales

group by pizza\_category

order by PCT\_category desc;



Calculates total revenue per pizza\_category and its **percentage share of overall revenue**, then lists results by category.

5. % of Sales by Pizza Size

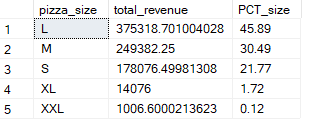
select pizza\_size, SUM(total\_price) as total\_revenue,

cast((SUM(total\_price) \* 100) / (select sum(total\_price) from pizza\_sales) as decimal(10,2)) as PCT\_size

from pizza\_sales

group by pizza\_size

order by PCT\_size desc;



Calculates total revenue per pizza\_size and its **percentage of overall revenue**, then orders results by pizza size.

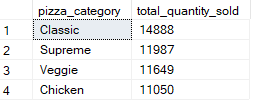
6. Total Pizzas Sold by Pizza Category

select pizza\_category, sum(quantity) as total\_quantity\_sold

from pizza\_sales

group by pizza\_category

order by total\_quantity\_sold desc;



Sums quantity per pizza\_category to get **total pizzas sold by category**, then orders results from highest to lowest.

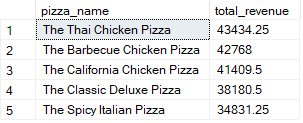
7. Top 5 Pizzas by Revenue

select top 5 pizza\_name, round(sum(total\_price), 2) as total\_revenue

from pizza\_sales

group by pizza\_name

order by total\_revenue desc;



Finds the **top 5 pizzas generating the highest revenue** by summing total\_price for each pizza\_name and sorting in descending order.

8. Bottom 5 Pizzas by Revenue

select top 5 pizza\_name, round(sum(total\_price), 2) as total\_revenue

from pizza\_sales

group by pizza\_name

order by total\_revenue asc;



Finds the **5 pizzas with the lowest revenue** by summing total\_price per pizza\_name and sorting in ascending order.

9. Top 5 Pizzas by Quantity

select top 5 pizza\_name, sum(quantity) as total\_pizzas\_sold

from pizza\_sales

group by pizza\_name

order by total\_pizzas\_sold desc;



Retrieves the **5 most sold pizzas** by summing quantity for each pizza\_name and ordering results from highest to lowest.

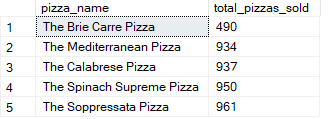
10. Bottom 5 Pizzas by Quantity

select top 5 pizza\_name, sum(quantity) as total\_pizzas\_sold

from pizza\_sales

group by pizza\_name

order by total\_pizzas\_sold asc;



Retrieves the **5 least sold pizzas** by summing quantity per pizza\_name and ordering results from lowest to highest.

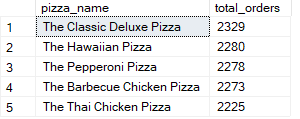
11. Top 5 Pizzas by Total Orders

select top 5 pizza\_name, COUNT(distinct order\_id) as total\_orders

from pizza\_sales

group by pizza\_name

order by total\_orders desc;



Finds the **5 most frequently ordered pizzas** by counting unique order\_id values per pizza\_name, sorted from highest to lowest.

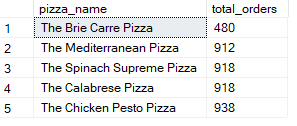
12. Bottom 5 Pizzas by Total Orders

select top 5 pizza\_name, COUNT(distinct order\_id) as total\_orders

from pizza\_sales

group by pizza\_name

order by total\_orders asc;



Finds the **5 least frequently ordered pizzas** by counting unique order\_id values per pizza\_name, sorted from lowest to highest.