**SQL Queries for Analysis**

-- 1. Total Revenue Generated Each Month

-- This query calculates the total revenue generated for each month.

-- It groups the orders by the month of the order date and sums up the sales for each group.

-- The results are ordered by the month for clarity.

SELECT

MONTH(order\_date) AS month,

SUM(sales) AS total\_revenue

FROM

processed\_orders

GROUP BY

MONTH(order\_date)

ORDER BY

month;

-- 2. Average Order Value by Customer

-- This query calculates the average order value for each customer.

-- It groups the orders by customer ID and computes the average sales per customer.

SELECT

customer\_id,

AVG(sales) AS avg\_order\_value

FROM

processed\_orders

GROUP BY

customer\_id;

-- 3. Number of Orders Placed on Weekends

-- This query counts the number of orders that were placed on weekends.

-- It filters the orders by checking if the order's weekday is either Saturday (6) or Sunday (7).

SELECT

COUNT(order\_id) AS weekend\_orders

FROM

processed\_orders

WHERE

order\_weekday IN (6, 7);

-- 4. Product Category with the Highest Average Sales

-- This query identifies the product category with the highest average sales.

-- It groups the orders by category ID, calculates the average sales for each category,

-- and orders the results in descending order, limiting the output to the top category.

SELECT

category\_id,

AVG(sales) AS avg\_sales

FROM

processed\_orders

GROUP BY

category\_id

ORDER BY

avg\_sales DESC

LIMIT 1;

-- 5. Distribution of Order Counts per Customer

-- This query shows the distribution of order counts among customers.

-- It groups customers by their total number of orders and counts how many customers fall into each group.

SELECT

customer\_order\_count,

COUNT(\*) AS num\_customers

FROM

processed\_orders

GROUP BY

customer\_order\_count

ORDER BY

customer\_order\_count;

-- 6. Total Sales by Weekday

-- This query calculates the total sales made on each weekday.

-- It groups the orders by the weekday of the order date and sums the sales for each day.

SELECT

order\_weekday,

SUM(sales) AS total\_sales

FROM

processed\_orders

GROUP BY

order\_weekday

ORDER BY

order\_weekday;

-- 7. Top 10 Customers by Total Sales

-- This query finds the top 10 customers based on their total sales.

-- It groups the orders by customer ID, sums the sales for each customer,

-- and orders the results in descending order, limiting the output to the top 10 customers.

SELECT

customer\_id,

SUM(sales) AS total\_sales

FROM

processed\_orders

GROUP BY

customer\_id

ORDER BY

total\_sales DESC

LIMIT 10;

-- 8. Average Sales per Order for Each Month

-- This query calculates the average sales per order for each month.

-- It groups the orders by the month of the order date and calculates the average sales for each group.

SELECT

MONTH(order\_date) AS month,

AVG(sales) AS avg\_sales\_per\_order

FROM

processed\_orders

GROUP BY

MONTH(order\_date)

ORDER BY

month;

-- 9. Number of Orders Placed During Each Hour of the Day

-- This query counts the number of orders placed during each hour of the day.

-- It groups the orders by the hour of the order time and counts how many orders were placed in each hour.

SELECT

order\_hour,

COUNT(order\_id) AS num\_orders

FROM

processed\_orders

GROUP BY

order\_hour

ORDER BY

order\_hour;

-- 10. Total Revenue Generated by Each Product Category

-- This query calculates the total revenue generated by each product category.

-- It groups the orders by category ID and sums the sales for each category,

-- ordering the results in descending order to highlight the highest-grossing categories.

SELECT

category\_id,

SUM(sales) AS total\_revenue

FROM

processed\_orders

GROUP BY

category\_id

ORDER BY

total\_revenue DESC;

-- 11. Day of the Week that Generates the Most Revenue

-- This query identifies the day of the week that generates the most revenue.

-- It groups the orders by the weekday of the order date and sums the sales for each day,

-- ordering the results in descending order and limiting the output to the top day.

SELECT

order\_weekday,

SUM(sales) AS total\_revenue

FROM

processed\_orders

GROUP BY

order\_weekday

ORDER BY

total\_revenue DESC

LIMIT 1;

-- 12. Average Number of Orders per Customer

-- This query calculates the average number of orders placed by each customer.

-- It computes the average of the total number of orders placed by each customer.

SELECT

AVG(customer\_order\_count) AS avg\_orders\_per\_customer

FROM

processed\_orders;

-- 13. Products with the Highest Number of Orders

-- This query identifies the products with the highest number of orders.

-- It groups the orders by product ID and counts the number of orders for each product,

-- ordering the results in descending order and limiting the output to the top 10 products.

SELECT

product\_id,

COUNT(order\_id) AS num\_orders

FROM

processed\_orders

GROUP BY

product\_id

ORDER BY

num\_orders DESC

LIMIT 10;

-- 14. Correlation Between Order Time (Hour) and Sales

-- This query examines the correlation between the hour of the order time and the average sales.

-- It groups the orders by the hour of the order time and calculates the average sales for each hour.

SELECT

order\_hour,

AVG(sales) AS avg\_sales

FROM

processed\_orders

GROUP BY

order\_hour

ORDER BY

order\_hour;

-- 15. Number of Orders Placed by Customers Who Have Placed More Than 10 Orders

-- This query counts the number of orders placed by customers who have placed more than 10 orders.

-- It filters the orders to include only those customers and counts the total number of their orders.

SELECT

COUNT(order\_id) AS num\_orders

FROM

processed\_orders

WHERE

customer\_order\_count > 10;