**Analytical SQL Business Queries**

-----> Query 1<------

WITH cte AS (SELECT country, FORMAT(country\_revenue, '#,###.##') AS revenue,

RANK() OVER (ORDER BY country\_revenue DESC) AS country\_rnk

FROM ( SELECT DISTINCT country,

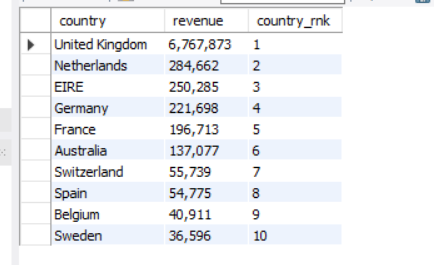
ROUND(SUM(quantity \* price) OVER (PARTITION BY country)) AS country\_revenue

FROM table\_retail ) x)

SELECT \* FROM cte

WHERE country\_rnk <= 10;

This query retrieves the top 10 countries with the highest total revenue based on the sales data stored in the online retail dataset, This query can help business stakeholders identify which countries are the most profitable in terms of retail sales and can inform strategic decisions around targeting those markets for growth.



-----> Query 2 <------

select StockCode , FORMAT(product\_sales, '#,###.##') product\_sales ,rank() over(order by product\_sales desc) as product\_rnk

from (select distinct StockCode ,

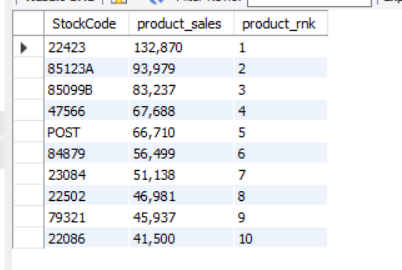
round(sum(quantity \* price) over(partition by StockCode)) as product\_sales

from table\_retail) x

order by product\_rnk

limit 10;

The business meaning behind this query is to help identify the top-selling products in the company's retail business. This information can be used to make decisions about inventory management, pricing, and marketing strategies. By focusing on the products that generate the most revenue, the company can optimize its operations and increase its profitability.

****

-----> Query 3 <------

select distinct StockCode , round(sum(quantity ) over(partition by StockCode)) as Quantity\_sold

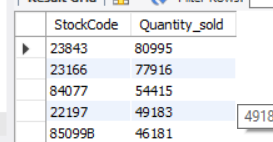
from table\_retail

where quantity > 0

order by Quantity\_sold desc

limit 5;

this simple query is to identify the products that have sold the most quantity in the given period.



-----> Query 4<------

with cte as (select month,total\_sales, lead(total\_sales) over(order by month desc) as last\_month\_sales from

(select distinct Extract(month from invoicedate) as month,

round(sum((price \* quantity)) over(partition by Extract(month from invoicedate))) total\_sales

from table\_retail

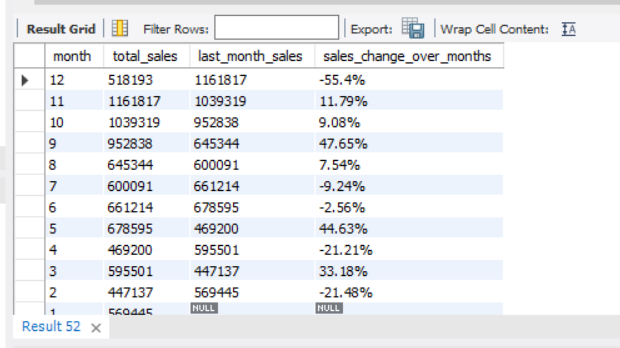
where quantity > 0 and Extract(year from invoicedate) = '2011' ) y

order by month desc)

select \*,concat(round(((total\_sales - last\_month\_sales)/last\_month\_sales) \*100 ,2), '%') "sales\_change\_over\_months"

from cte;

This SQL query calculates the monthly sales change percentage for the year 2011 in the online retail business, By calculating the percentage change in sales from one month to the next, the query can help identify whether sales are increasing or decreasing over time. This information can be useful for forecasting future sales and making business decisions such as adjusting inventory levels, marketing efforts, and production plans.



-----> Query 5<------

WITH customer\_sales AS (

SELECT

customer\_id,

country,

ROUND(SUM(quantity \* price), 2) AS total\_sales

FROM table\_retail

GROUP BY customer\_id, country

),

ranked\_sales AS (

SELECT

customer\_id,

country,

total\_sales,

ROW\_NUMBER() OVER(PARTITION BY country ORDER BY total\_sales DESC) AS sales\_rank

FROM customer\_sales

)

SELECT

country,

customer\_id,

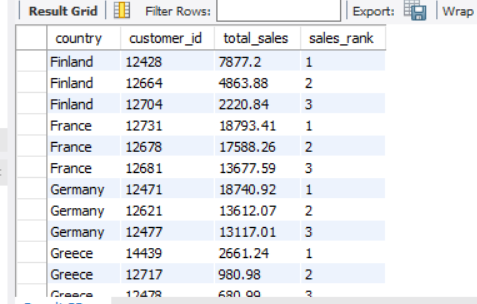
total\_sales, sales\_rank

FROM ranked\_sales

WHERE sales\_rank <= 3

ORDER BY country, total\_sales DESC;

This query selects the top 3 customers for each country based on their sales rank and returns their country, customer ID, and total sales.



-----> Query 6<------

SELECT customer\_id, total\_sales

FROM (

SELECT customer\_id, ROUND(SUM(quantity \* price)) AS total\_sales,

NTILE(5) OVER (ORDER BY ROUND(SUM(quantity \* price)) DESC) AS percentile

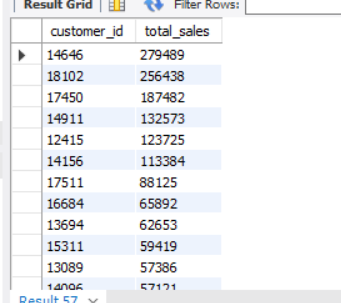
FROM table\_retail

group by customer\_id

) AS x

WHERE percentile = 1;

This query divides the customers into five groups based on their total sales, with group 1 containing the customers with the highest total sales. The outer query then selects only the customers in group 1, which represents the top 20% of customers according to total sales.



-----> Query 7<------

select distinct country , year, count(\*) over(partition by year, country) number\_of\_orders

from (select distinct invoice , Extract(year from cast(invoicedate as date)) as year , country

from table\_retail) X

order by 1 , 2;

This query will give us the count of total orders for each country for each year

