**1-Total number of transactions: This query gives us the total number of transactions made in the retail business.**

SELECT COUNT(DISTINCT Invoice) as Total\_Transactions FROM tableRetail;

**Business Meaning: This query helps us understand the volume of transactions made by the retail business. It is important to keep track of the number of transactions made to analyze the overall performance of the business.**

**2-Most popular products: This query provides the most popular products sold by the retail business based on the quantity sold.**

SELECT StockCode, Total\_Quantity\_Sold

FROM (

SELECT StockCode, SUM(Quantity) as Total\_Quantity\_Sold,

ROW\_NUMBER() OVER (ORDER BY SUM(Quantity) DESC) as row\_num

FROM tableRetail

GROUP BY StockCode

) t

WHERE row\_num <= 10;

***Business Meaning:* This query helps us identify the most popular products sold by the retail business. This information can be used to understand customer preferences and to optimize inventory management.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**3- Which month generated the most revenue?**

SELECT TO\_CHAR(TO\_DATE(InvoiceDate, 'MM/DD/YYYY HH24:MI'), 'YYYY-MM') AS Month, SUM(Price\*Quantity) AS Revenue

FROM tableRetail

GROUP BY TO\_CHAR(TO\_DATE(InvoiceDate, 'MM/DD/YYYY HH24:MI'), 'YYYY-MM')

ORDER BY Revenue DESC;

***Business Meaning*: this query is to identify the month that generated the most revenue. This information can be used to analyze sales trends and identify any patterns that may have contributed to higher revenue in that month. It can also help businesses to plan for future sales and marketing strategies to increase revenue during slower months.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**4- total sales for every stock code :**

SELECT DISTINCT

stockcode,

SUM(quantity \* price) OVER (PARTITION BY stockcode) AS total\_sales

FROM

tableRetail

ORDER BY

total\_sales DESC;

***Business Meaning*: The output of this query can provide valuable business insights for the company. For instance, the company can identify the most popular products by looking at the total sales figures for each stock code. They can use this information to optimize their inventory management by stocking up on popular products to meet the demand. Additionally, the company can analyze the total sales figures to determine the profitability of each stock code and use this information to make informed decisions about product pricing and promotions.**

**5- What is the average price of each item sold?**

SELECT distinct

StockCode,

AVG(Price) OVER (PARTITION BY StockCode) AS AveragePrice

FROM

tableRetail;

***Business Meaning*: This information can be useful for business analysts to understand the pricing trends for each product in the company's inventory, which can help them make informed decisions about pricing and inventory management.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Q3\_**

**a-**

SELECT

cust\_id,

MAX(num\_consecutive\_days) AS max\_num\_consecutive\_days

FROM (

SELECT

cust\_id,

calendar\_dt,

CASE

WHEN calendar\_dt = LAG(calendar\_dt) OVER (PARTITION BY cust\_id ORDER BY calendar\_dt) + 1 OR

LAG(calendar\_dt) OVER (PARTITION BY cust\_id ORDER BY calendar\_dt) IS NULL

THEN SUM(1) OVER (PARTITION BY cust\_id ORDER BY calendar\_dt)

ELSE 1

END AS num\_consecutive\_days

FROM mytable

) t

GROUP BY cust\_id;

b-

SELECT

Cust\_Id,

COUNT(\*) AS num\_transactions,

MAX(Calendar\_Dt) - MIN(Calendar\_Dt) + 1 AS num\_days,

MAX(total) AS final\_total

FROM (

SELECT

Cust\_Id,

Calendar\_Dt,

SUM(Amt\_LE) OVER (PARTITION BY Cust\_Id ORDER BY Calendar\_Dt) AS total

FROM mytable

)

WHERE total >= 250

GROUP BY Cust\_Id;