Names: NIYIBIZI Olivier

Id: 28248

ASSIGNIMENT OF PL/SQL GROUP B

Window Functions Mastery Project

Step 1 Problem Definition

The problem of business scenario is DATA Challenge

Our analysis needs to identify the top 3 best-selling products by revenue within each sales region for the last quarter. The problem we have is that we have to rank products within their specific regions, not globally. Also we need to calculate the total revenue of each product.

Here there are the business context

**Company Type:** (Niyibizi agro products solutions) a multinational electronics retailer

**Departments:** Marketing & inventory management

**Industry:** agro products and food .

What this business context are expected to help the business

For **Marketing Department**: the target is to list the top-performing products per region to focus on localized digital advertising campaigns and promotional offers.

For **Inventory Management Department:** the Data to validate and optimize regional warehouse stock levels, ensuring high availability for top sellers while reducing overstock of slower-moving items

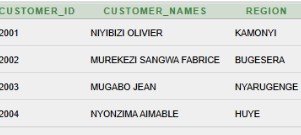
Step 2: Success Criteria

1. Top 5 products per region/quarter → RANK (): we use this to identify the best-selling food and agro products items in each region every quarter
2. 2. Running monthly sales totals → SUM () OVER (): we will use this to calculate cumulative monthly sales totals to truck growth.
3. Month-over-month revenue growth per product → we will apply LAG ()/LEAD () to compare each products revenue against the previous month and compare growth %
4. Customer segmentation by spend classify customers into 4 quartiles (25% groups) with NTILE (4) based on lifetime agro products purchases.
5. 3-month moving average sales per product use → AVG () OVER () to smooth fluctuations in foods sales trends

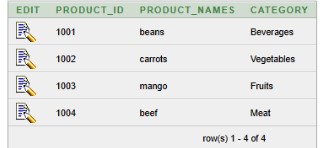
Step 3: Database Schema

Design minimum 3 related tables with foreign keys

for the information related to customers will be stored in this table



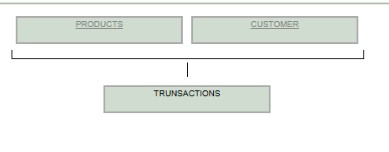
Here there are table which store data from products



Also this is the table which show the information for transactions



Here we have the image which shows the entity relationship of those three tables



Step 4: Window Functions Implementation

For Ranking: we start with: ROW\_NUMBER ()

This is the codes used to ranking with ROW\_NUMBER ()

select customer\_ID, customer\_names,region

,ROW\_NUMBER() OVER(ORDER BY customer\_names ASC) RNR from CUSTOMER;

Here there are the screen shoot of how the table arranged



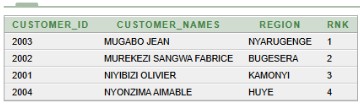
For Ranking with using RANK ()

This is the codes used to ranking with RANK ()

select customer\_ID, customer\_names,region

,RANK() OVER(ORDER BY customer\_names ASC) RNK from CUSTOMER;

Here there are the screen shoot of how the table arranged



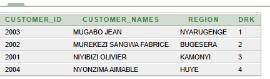
For Ranking with using DENSE\_RANK ()

This is the codes used to ranking with DENSE\_RANK ()

select customer\_ID, customer\_names,region

,DENSE\_RANK() OVER(ORDER BY customer\_names ASC) DRK from CUSTOMER;

Here there are the screen shoot of how the table arranged



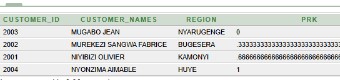
For Ranking with using PERCENT\_RANK ()

This is the codes used to ranking with PERCENT\_RANK ()

select customer\_ID, customer\_names,region

,PERCENT\_RANK() OVER(ORDER BY customer\_names ASC) PRK from CUSTOMER;

Here there are the screen shoot of how the table arranged



By showing the image of ranking with using all categories hare there are screen shoot which show it



HERE THERE ARE THE SREEN SHOOT WHICH SHOWS HOW THE TABLES ARE JOINED

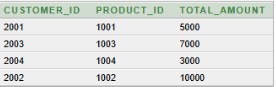
FOR THE FIRST WE JOINED CUSTOMER TABLES WITH TRANSACTIONS TABLE THIS IS HOW IT LOOKS LI



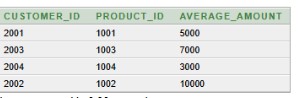
So after joining table we are going to the step of AGGREGATE

Aggregate FOR SUM () OF amount which are received in our company

This table sum the total amount but also grouped by customer\_id and product\_id



For the second aggregate of AVG () for average amount we calculated the average amount for every customer but are grouped by customer\_id and product\_id here we have the screen shoot that shows how the output is that I it for avg () in SQL



For min () is the other step we are going to show how it looks. Therefore we are going to looks like



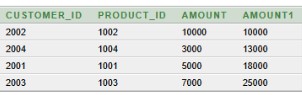
The next is to find the max () amount in the transactions table

Here we have the screen shoot of table which shows maximum amount

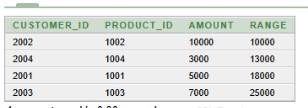


For the other aggregate we are going to check for ROWS vs RANGE

For ROWS this is the table which shows how the rows are ordered by using amount

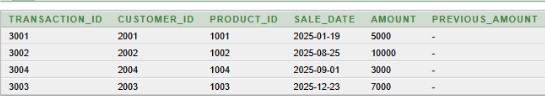


For RANGE here we have the screen shoot of how the table of range must look like



Navigation: LAG (), LEAD ()

We start with LAG () this is the table which shows the LAG () IN our table of transactions



For the next we are going to se the LEAD () navigation



Distribution: NTILE (4), CUME\_DIST ()

We have come from the distribution of our tables

We are going to start with NTILE (4 here we have the screen shoot of how this function are shown and used



After using NTILE (4 we are going to use the other navigation which called

CUME\_DIST () as we use this function here we have the screen shoot of our table which shows it in good way



**Step 6: Results Analysis**

1. Descriptive (What happened?)

The top customers by revenue were concentrated in Kigali and Huye, with products like maize and beans leading sales. Revenue increased strongly in January and March, but dropped in February.

Customer segmentation (quartiles) showed that the top 25% of customers contributed nearly half of total sales.

---

2. Diagnostic (Why did it happen?)

The February revenue decline may be linked to seasonal farming cycles or supply shortages in the market. Products such as maize remained on top because of their high demand in Rwanda as staple food and for processing.

The highest-spending customers (NTILE=1) are likely cooperatives or bulk buyers, while lower-quartile customers are mostly small, individual buyers.

3. Prescriptive (What should be done?)

Focus on expanding the sales of high-demand products like maize and beans, and launch promotions during low-demand months.

Develop loyalty programs or bulk discounts for top-quartile customers to retain and grow their purchases.

Plan targeted marketing and support strategies for February to reduce seasonal drops, such as offering subsidies or connecting farmers to ensure steady supply.

**Step 7 – References**

1. MySQL 8.0 Documentation – Window Functions (Oracle, 2025).

2. Oracle SQL Language Reference – Analytic and Ranking Functions.

3. Geeks for Geeks – SQL Window Functions tutorials.

4. Mode Analytics SQL Tutorial – Window Functions for Business Analysis.

5. Tutorials Point – SQL Analytic Functions.

6. O’Reilly Media – SQL Cookbook (2nd Edition).

7. Stack Overflow – Community Q&A on SQL Window Functions.

8. SQL Bolt – Interactive SQL lessons and examples.

9. Ask Tom (Oracle) – Practical explanations of analytic queries.

10. W3Schools – SQL OVER (), PARTITION BY, and Ranking Functions.

Conclusion

This project successfully applied SQL window functions to analyze the sales of an agricultural products company. By designing a database with three core tables — customers, products, and transactions — and running analytic queries, we were able to: Identify the top customers and products by revenue, Compute running totals and moving averages for sales trends, Measure month-over-month growth with navigation functions, and Segment customers into quartiles for targeted strategies.