Individual Assignment I: SQL JOINs & Window Functions Project

Course: Database Development with PL/SQL (INSY 8311) **Instructor:** Eric Maniraguha | eric.maniraguha@auca.ac.rw **Assignment Date:** September 19, 2025 **Groups:** A, B, C, D **Deadline: Firday, February 06, 2025 (12:00 PM)** — **No late submissions accepted**

Message to Students

This assignment is critical for your learning and future assessments. The SQL concepts and implementations you master here will appear in **upcoming quizzes and the midterm examination**.

To Maximize Your Grade

1. **Read and follow all instructions carefully**
2. **Use the required formats exactly**
3. **Submit a complete project on time** (late submission = zero)
4. **Provide clear analysis and professional documentation**

Academic Integrity Guidelines

- Read the instructions carefully.
- Prepare your work clearly and professionally.
- **Avoid copy-paste** from online sources or other students — this is plagiarism.
- **Do not use ChatGPT or any other AI tool** to generate answers. If detected, you will receive **zero marks for the entire assignment**.
- Paraphrase and express ideas in your own words.

Assignment Overview

Objective

Demonstrate practical mastery of:

- **SQL JOINs** for multi-table relational analysis
- **SQL/PL/SQL Window Functions** for analytical and business reporting

You will solve a realistic business problem, implement queries, and document your work in a **professional GitHub repository**.

> **Note:** Students may use **any DBMS tool** (Oracle, PostgreSQL, MySQL, SQL Server, etc.) for this project.

Weight

10 raw points (aggregated to **5 points** in the final grade)

Assignment Structure

This assignment has **two main technical parts**:

- **Part A:** SQL JOINs
- **Part B:** SQL Window Functions

Both parts must use the **same database schema and business scenario**.

Step-by-Step Requirements

Step 1: Problem Definition (2 pts)

Define a **specific and measurable business scenario**.

Required Format

- **Business Context** (company type, department, industry)
- **Data Challenge** (2–3 sentences explaining the problem)
- **Expected Outcome** (decision or insight expected from analysis)

High scoring example:

> Identify top products per region, analyze customer purchasing frequency, and segment customers for targeted marketing.

Low scoring example:

> Analyze sales data to find patterns.

Step 2: Success Criteria (part of 2 pts)

Define **exactly five (5) measurable goals**, clearly linked to window functions:

1. Top 5 products per region or quarter → `RANK()`
2. Running monthly sales totals → `SUM() OVER()`
3. Month-over-month growth → `LAG()` / `LEAD()`
4. Customer quartile segmentation → `NTILE(4)`
5. Three-month moving averages → `AVG() OVER()`

Step 3: Database Schema Design

Design **at least three (3) related tables** with primary and foreign keys.

An ER diagram is mandatory.

Step 4: Part A — SQL JOINs Implementation

Demonstrate **correct and meaningful use of SQL JOINs** using the tables from Step 3.

Required JOIN Types

Each student must implement **all** of the following:

1. **INNER JOIN** *Retrieve transactions with valid customers and products*
2. **LEFT JOIN** *Identify customers who have never made a transaction*
3. **RIGHT JOIN** *(or FULL JOIN if RIGHT JOIN is avoided) Detect products with no sales activity*
4. **FULL OUTER JOIN** *Compare customers and products including unmatched records*
5. **SELF JOIN** *Compare customers within the same region or transactions within the same time period*

Required Format (for each JOIN)

- SQL query with comments
- Screenshot showing results
- Business interpretation (2–3 sentences)

Step 5: Part B — Window Functions Implementation

Implement **all four categories** of window functions (**1 point per category**).

Required Categories

1. **Ranking Functions** `ROW_NUMBER()`, `RANK()`, `DENSE_RANK()`, `PERCENT_RANK()` *Use case: Top N customers or products by revenue*
2. **Aggregate Window Functions** `SUM()`, `AVG()`, `MIN()`, `MAX()` Use both `ROWS` and `RANGE` frames *Use case: Running totals and trends*
3. **Navigation Functions** `LAG()`, `LEAD()` *Use case: Period-to-period comparison and growth*
4. **Distribution Functions** `NTILE(4)`, `CUME_DIST()` *Use case: Customer segmentation*

Required Format (for each function)

- SQL query with comments
- Screenshot (results clearly visible)
- Interpretation (2–3 sentences)

Step 6: GitHub Repository

- **Repository name:** `plsql_window_functions_[studentId]_[firstname]`
- **Visibility:** Public

Repository Must Contain

- SQL scripts (error-free)
- Screenshots (clear and well organized)
- Professional `README.md` including:
    - Business problem
    - Schema and ER diagram
    - JOIN queries
    - Window function queries
    - Key insights
    - References
    - Integrity statement
    - **Selected screenshots** proving personal work

Step 7: Results Analysis

Provide insights at **three analytical levels**:

1. **Descriptive** — What happened?
2. **Diagnostic** — Why did it happen?
3. **Prescriptive** — What should be done next?

Step 8: References

- Add **references**
- Any citation style is acceptable
- Include official documentation, tutorials, academic or business resources

Add the following statement in the README:

> "All sources were properly cited. Implementations and analysis represent original work. No AI-generated content was copied without attribution or adaptation."

Grading Breakdown (10 pts → 5 pts final)

- Problem Definition & Success Criteria
- SQL JOINs Implementation
- Window Functions Implementation
- Results Analysis
- Technical Quality, GitHub & Integrity

Submission Instructions

To: eric.maniraguha@auca.ac.rw **Subject:** `INSY 8311 SQL Assignment I – [Full Name] – Group [X]`

Email Body:

```
Repository Link: [GitHub URL]
Business Problem: [1 sentence]
Key Findings: [2 insights]
Sources Consulted: [number]
```

Final Checklist

- ☐ Repository is public and accessible
- ☐ SQL runs without errors
- ☐ Screenshots included
- ☐ README is clear and professional
- ☐ JOINs and Window Functions completed
- ☐ References and integrity statement included

Professional & Ethical Note

> *"Whoever is faithful in very little is also faithful in much."* — Luke 16:10

As database professionals, your reputation depends on **accuracy, integrity, and responsibility**.
