## Instructions:

* The assignment aims to reinforce SQL query skills using the **WideWorldImporters** database. You will write and execute SQL queries to retrieve and manipulate data using various SQL constructs covered in Chapter 4 of our SQL text.
* Write SQL queries that fulfill the requirements listed below.
* Use appropriate column names, functions, and sorting techniques where necessary.
* Ensure queries return meaningful results based on the dataset.
* Save your work as a .sql file and upload it to D2L Dropbox folder for the assignment. Use the following file naming convention: *yourFirstName\_yourLastName\_SQL\_HW\_03.sql*

1. Retrieve the total number of customers, average credit limit, maximum credit limit, and minimum credit limit from the Sales.Customers table.
2. List the total sales amount per customer from the Sales.Invoices table. Group the results by CustomerID.
   1. Use **ExtendedPrice** from Sales.InvoiceLines to calculate sales amounts.
3. Retrieve customer IDs with total sales exceeding $50,000 from the Sales.Invoices table. Use GROUP BY and HAVING.
4. Task:
   1. Write a query that lists invoices where TotalDryItems > 20 using the WHERE clause.
   2. Modify the query to use HAVING after grouping by CustomerID.
5. Retrieve sales records from Sales.Invoices where the TotalDryItems > 50 or the TotalExcludingTax is between $5,000 and $10,000, and the InvoiceDate is in 2016.
6. Using the Sales.Invoices table, calculate the total sales by CustomerID and InvoiceYear with a grand total. Use the ROLLUP operator.
7. Generate a sales summary showing total sales by CustomerID and InvoiceYear, including all possible subtotal combinations using the CUBE operator.
8. Produce a report with the total sales by CustomerID, by InvoiceYear, and a grand total using GROUPING SETS.
9. Show each invoice’s TotalExcludingTax and the average sales amount over all invoices using the OVER() clause in the Sales.Invoices table.
10. Retrieve each invoice with its total sales and the average sales per CustomerID using OVER(PARTITION BY CustomerID).

**Grading Rubric**

Each query is worth \*\*10 points\*\*, evaluated based on the following criteria:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criteria** | **Excellent (10 - 9 pts)** | **Good (8 - 7 pts)** | **Satisfactory (6 - 5 pts)** | **Needs Improvement (4 - 0 pts)** |
| **Correctness of Query (50%)** | Query executes without errors and returns the expected result set. | Query executes with minor logical errors but mostly correct. | Query contains errors affecting correctness. | Query does not execute properly or is incorrect. |
| **Use of Required SQL Concepts (30%)** | Successfully applies the required SQL concepts. | Uses most required SQL concepts correctly. | Uses some SQL concepts but omits key elements. | SQL concepts are misused or omitted. |
| **Query Formatting & Readability (10%)** | Query is well-structured and properly formatted. | Query is readable but could use better formatting. | Query is difficult to read with inconsistent formatting. | Query lacks readability and proper structure. |
| **Use of Proper Column Naming and Aliasing (10%)** | All columns are named meaningfully. | Most columns are appropriately named. | Some columns are meaningfully named. | Column naming and aliasing are unclear or missing. |