## 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 7 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\_05.sql*

**Task 1: Create a Test Table**

CREATE TABLE TestCustomers (  
 CustomerID INT PRIMARY KEY IDENTITY(1,1),  
 CustomerName NVARCHAR(100) NOT NULL,  
 BuyingGroupID INT NULL,  
 PaymentDays INT NULL,  
 CreditLimit DECIMAL(18,2) NULL  
);

**Task 2: Use SELECT INTO to Copy Data**

**Description:** Copy a subset of customers from **Sales.Customers** into **TestCustomers**, only those with a **CreditLimit** greater than 2000.

**Task 3: Insert a Single Row**

**Description:** Insert a single new customer into **TestCustomers**.

**Task 4: Insert Multiple Rows**

**Description:** Insert three new customers into **TestCustomers**.

**Task 5: Insert Rows with Default and NULL Values**

**Description:** Insert a new row with **NULL values** and **default values** for specific columns.

**Task 6: Insert Rows Selected from Another Table**

**Description:** Insert customers from **TestCustomers\_Copy** into **TestCustomers**.

**Task 7: Perform an Update Operation**

**Description:** Update **CreditLimit** for customers with a **NULL** value, setting it to 5000.

**Task 8: Use a Subquery in an Update Operation**

**Description**: Increase the **CreditLimit** of customers in **TestCustomers** by 10% if their **BuyingGroupID** matches an existing **BuyingGroupID** in **Sales.Customers**.

**Task 9: Delete Rows Using a Subquery and Joins**

**Description:** Delete customers from **TestCustomers** who are not in **TestCustomers\_Copy**.

**Task 10: Perform a MERGE Operation**

**Description:** Merge data from **TestCustomers\_Copy** into **TestCustomers**, updating **CreditLimit** if a customer exists, inserting a new row otherwise.

**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. |