Name: Ronnie Sokha

Date: 8/20/23

Course: Foundations of Databases & SQL Programming

Assignment 07: Functions

GitHub Link: https://github.com/RSokha/DBFoundations-Module07

SQL Functions

Introduction

Functions are useful for performing repetitive actions that you are going to use across numerous queries. There are default functions, but a user can create a function to perform specific actions and return desired results. This is called a User-Defined Function (UDF). What's great about a function is that it can be created once and called any amount of times in your script. There are different types of UDFs – each serving a different purpose and can be used on an ad-hoc scenario basis.

When to use a UDF

A User-Defined Function (UDF) is a function created by a user. These can simply be a function that accepts a single or multiple value that returns one output. There are default functions that live in Microsoft SQL Server that are commonly used – these are

- MIN()
- MAX()
- COUNT()
- SUM()
- AVG()

The benefit of a UDF is that a user can create a function to query and obtain specific information. For instance, if a company is looking at KPI metrics across a sales team and want to examine daily metrics for particular quotas and goals. A user can utilize a default function or a UDF to aggregate, calculate, and return an output.

Differences – Scalar, Inline, and Multi-Statement Functions

A Scalar function is a UDF that returns a single value each time it is executed. In other words, a scalar function performs calculations on inputs and parameters and returns a single output. Similarly, an Inline function also takes inputs however, instead of returning a single scalar value, it returns a single result from a single table. A Multi-Statement function is an essentially an extension of an Inline function. That is, it takes an input and returns a single result but it can be made from multiple statements. It is

important to mention that a Multi-Statement function requires a Begin and End block as well as defining the structure of the table.

Summary

UDFs are powerful tools that are easily available at a user's disposal. Data transformation and reference value retrieval are common uses for functions. What's interesting about UDFs in SQL isn't *what* they can do, it's *where* they can do it. The fact that they're executed in the database is what makes them useful. For instance, if we want to perform complex calculations on a certain set of input values, we can use functions to 'do the math' and return a single output. Additionally, we can store this particular function on this calculation and call it later in our program. This is just a simple example of a function and how it can be useful to a user.