Pauline Baculi
5/31/2022
IT FDN 130 A Sp 22
Assignment 07
https://github.com/pbacul/DBFoundations-Module07

SQL User Defined Functions

Introduction

In this assignment, I will explain SQL user defined functions (UDFs) and when they would be used. And within user defined functions there are scalar, inline, and multi statement functions, which I'll explain the differences of as well.

When to use a SQL UDF

A SQL user defined function (UDF) is a custom function that can be created and saved in a database. It is complex, robust query that is saved for repeated actions. It is useful in that it can be created once and then used multiple times afterwards without having to rewrite the entire query every time. Additionally, it can be used to format the data the way you want it presented and make it look cleaner.

The differences between Scalar, Inline, and Multi-Statement Functions

There are 3 different types of user defined functions: scalar, inline table valued, and multi-statement table valued.

A scalar UDF returns a single value/data point. It is not treated like a table when placed in a SELECT statement, but rather one would indicate SELECT and then the 2-part name of the scalar UDF. When writing a scalar UDF, the function name must be the 2-part name that includes the schema and have a Begin and End block. A scalar UDF can be a useful tool as a check constraint.

Unlike a scalar UDF, an inline table valued UDF returns results as a table of values and acts very similar to a table created in a database. When writing code to create a table valued UDF, the syntax used is "Returns Table As" and can only have one SELECT statement. Some additional differences from a scalar UDF: the function name does not need to be a 2-part name that includes the schema and a Begin and End block is not required.

A multi-statement table valued UDF returns the results as a table like an inline table valued UDF, but the difference is that it will perform multiple SELECT statements, instead of just one, and you can create the table and columns returned instead of already existing columns and tables. An additional difference from a scalar UDF is that a Begin and End block is required. When creating a multi-statement table valued UDF, it requires a table variable and includes the syntax "Returns @tablevariable Table."

Summary

In this assignment, I explained SQL user defined functions (UDFs) and when they are useful. I also explained the different types of UDFs (scalar, inline, and multi-statement functions) and their differences.