Name: Ron Klangnarong
Date: March 1, 2022
Course: Foundations of Databases and SQL Programming

Assignment 7 - Writeup for Module 7

GitHub URL: https://github.com/ronklnr/DBFoundations-Module07

Introduction:
This write-up is part of the assignment for Module 7. It answers two questions: when to use an SQL UDF and the differences between Scalar, Inline, and Multi-Statement Functions.

When to use an SQL UDF:
UDFs, or user-defined functions, allow users to create custom functions. These functions perform a set of defined actions, such as calculations or formatting, then return values. UDFs are not only useful for recurring tasks, but users can use them to transform data, check constraints, and select reporting data.

Differences between Scalar, Inline, and Multi-Statement Functions:
UDFs have three types: Scalar Valued Functions, Inline Functions, and Multi-Statement Functions. Scalar Valued Functions return a single value. Inline Functions and Multi-Statement Functions are table-valued functions. As the name suggests, they return a table of values. With Multi-Statement Functions, however, users can use multiple SELECT Statements to add actions.

There are a few differences in syntax. Users must specify the type of functions by writing "RETURNS (data type)" for Scalar Functions and "RETURNS TABLE" for Inline Functions. For Multi-Statement Functions, users have to declare a table variable (e.g. RETURNS @t TABLE). Unlike Inline Functions and Multi-Statement Functions, Scalar Values Functions require a BEGIN/END block between the SELECT Statement.

Summary:
With UDFs, or user-defined functions, users can repeatedly perform tasks without writing new code. UDFs can also be used for constraint checking and reporting purposes. Scalar Functions, one of the three UDF types, return a single values, while Inline and Multi-Statement Functions return a table of data.

Reference:
Gould, A. (2013, Feb 20), SQL Server Programming Part 10 - Table Valued Function, *WiseOwlTutorials*, www.youtube.com/watch?v=nCAEgNxC7nU