Amy Borgmeyer

August 24, 2020

IT FDN 130 A

Assignment 07

<https://github.com/AmyBo1123/DBFoundations-Module07>

Functions

# Introduction to functions

Information in databases is stored in tables but the process for joining and accessing data across tables can be laborious. Creating views, functions, and stored procedures expedites the retrieval of data by setting up defined code to access commonly needed views.

# When to Use a SQL User-Defined Function (UDF)

SQL Server has large number of built-in functions (summative, reformatting, and others). However, it is often more convenient, when a function will be used repeatedly or is complex to build and reference, to build a function that returns the specific desired value or table in the desired format:

* Scalar functions: Return a single value
* Table functions: Return a table of data based on defined parameters and instructions

Once a UDF is created, it can be managed in much the same way that other items are – via the create/alter/and drop functions.

# Differences between Scalar, Inline, and Multi-Statement Functions

There are multiple types of functions that serve different purposes.

* Scalar functions use one or more parameters to return a single value. Complex calculations based on multiple variables are commonly rendered as scalar functions.
* Inline functions are a type of table functions, but they are limited to a single select statement where the columns selected define the return table columns. Inline functions essentially function as a view.
* Multi-statement functions are another subset of table functions but they permit the most complex sets of instructions. In multi-statement functions, it is possible to execute multiple queries and aggregate the results into a single table of results.

# Summary

Creating functions that predefine commonly defined queries or result sets is a more efficient way of querying data and can save time and effort.