

Name: KGuite

Date: 2023-08-22

Course: IT FDN 130 A

Assignment Module07: Functions

Github: <https://github.com/kguite/DBFoundations-Module07>

Module 07: Functions

Introduction:

This week, we explored SQL built-in functions as well as creating our own (user-defined) functions.

When would you use a SQL UDF (User Defined Function)?

A User Defined Function in SQL has benefits just like functions in other languages. They accept parameters, perform an action, and return the result. The type of function you use depends on what result you're looking for and how you're using the function.

A few benefits of using SQL User Defined Functions are:

- Modular Programming: the function can be created once and called multiple times.
- Faster execution: operations can be cached, reducing execution times
- Reduce network traffic: since the function call can be invoked in a WHERE clause, rather than a long string of complex SELECT statements, the total of SQL lines sent to the client are reduced.

What are the differences between Scalar Functions, Inline Functions, and Multi-Statement Functions?

- Scalar Function: returns a single value
 - Example: a function that takes in a single value (ProductID), performs calculations, and returns a single value (sum).
- Inline Function: returns a table
 - Example: a function that takes one input parameter (StoreID) and returns a table that displays Products and Sales
- Multi-Statement Function: returns a table, giving the user who defines it control over the definition of the table variable, naming of the columns, setting up data types, constraints, etc. Parameters are common but not required.
 - Example: a function that takes one input (year) and returns all employees born that year.

Summary:

To conclude, there are many types of functions, both SQL-defined and user-defined, that we can use to return the information from the database that we are looking for.