Elizabeth Yeo

May 26, 2021

ITFnd130

Assignment 7

< https://github.com/Elizabeth8991/DBFoundations-Module7.git>

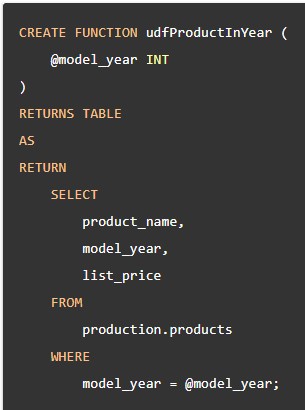
**SQL Functions**

**Introduction**

Functions in SQL can be simple or quite complex depending upon the data in the database that is being used. “Functions are methods used in data operations.” (edureka, [SQL Functions | Aggregate and Scalar Functions with Examples | Edureka](https://www.edureka.co/blog/sql-functions) 2021). There many different types of functions that allow us to do different things. There are Aggregate functions which perform calculations on a set of values and then return a singled value. However, as data becomes more complex, more complex and specialized functions are needed such as User Defined functions, Scaler functions, Inline functions, and Multi-Statement Functions.

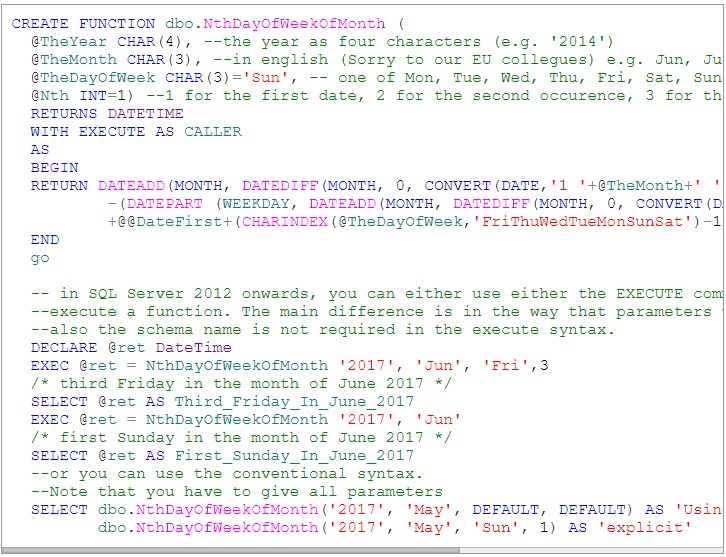
**When to Use a SQL User Defined Function**

In order to know when to use a SQL User Defined Function or UDF, it is important to know what a UDF is. A User Defined Function (UDF) “…is a programming construct that accepts parameters, does work that typically makes use of the accepted parameters, and returns a type of result.” (techrepublic, [Understand when to use user-defined functions in SQL Server - TechRepublic](https://www.techrepublic.com/blog/the-enterprise-cloud/understand-when-to-use-user-defined-functions-in-sql-server/#:~:text=In%20the%20simplest%20terms%2C%20a,%2Dvalued%20and%20scalar%2Dvalued.) 2021). There are also two types of a UDF: Table-Valued and Scaler-Valued. To be brief, a Table-Valued UDF is a function that accepts a set of parameters and returns the results in a table; you can also query that table and join it to other tables should you need to. (**Figure 1).**



***Figure 1: Example of a Table Valued Function in Context***

The Scaler-Valued UDF returns a single value after accepting a set of parameters. A UDF would be used when values from one or more tables needs to be used via a join to perform a calculation that returns an aggregate result. To put it another way, a View allows the user to see specific sets of data form multiple tables whereas a UDF is used to perform calculations based on parameters and returns a result. That is when a UDF would be best used. (**Figure 2).**

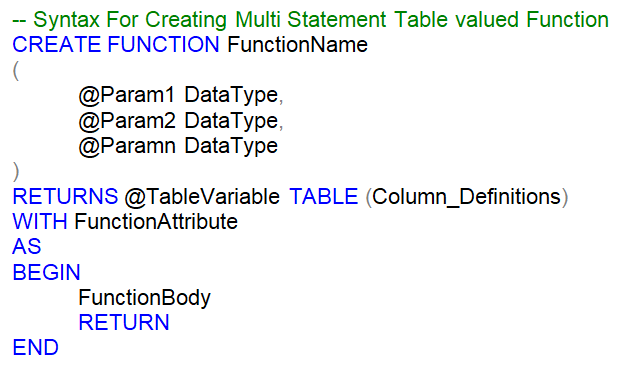
****

***Figure2: Example of a Complex Scaler Function in Context***

**Differences in Scaler, Inline, And Multi-Statement Functions**

As explained previously, the Scaler function returns a single value after accepting a set of parameters and executing the code. An Inline Function, however, “…is similar to a view. This means that an inline function can only contain a *single* SELECT statement, and the columns in the SELECT statement implicitly define the columns of the returned table set of the function.” (SQLSunday, [Table value functions vs Inline functions | sqlsunday.com](https://sqlsunday.com/2013/05/05/table-value-vs-inline-table-functions/#:~:text=If%20a%20table%20value%20function,table%20set%20of%20the%20function.) 2021).

A Multi-Statement Function is primarily used in Transact-SQL Statements. An example of this is when you purchase something on-line at a store. Even though the purchaser does not see the code in the background, the code is written in a way for that transaction to take place. The difference is how these functions are used and what is returned after they are executed in a block of code. (**Figure 3**)



***Figure 3: Multi-Function Syntax***

**Summary**

Functions allow database engineers to return values based on a set of parameters or conditions. Engineers can have single values returned, multiple values returned, or for transactions that take place in the world. Without functions, it would not be easy to manipulate the data in very necessary ways to either view or use the data that is returned.