IMarshall

11/29/2022

IT FDN 130

Assignment 07

https://github.com/IMarshallUW/DBFoundations-Module07

Functions

Introduction

In the last module we went over SQL Views and the similarities they have with SQL Functions. In this paper we will go over in functions in more detail. Particularly what a User Defined Function (UDF) is, when you would use one, and three types of functions.

UDF

To start we will discuss UDF's and the situations that would warrant using one. While there a great deal of functions in SQL server, SQL assistant, and many other programs used to write SQL that can perform helpful tasks. These tasks including, but not limited to, retreiving hidden date information, performing background mathmatical equations, replacing data on a mass scale, and a great deal more. We will sometimes find that we would like to create our own function to provide quick and accurate data for a user.

While views can still be used to present this information to be used by Tableau or some other data visualization tool, a UDF can give you the specific results quicker and easier for those who just want an answer. A great example is using sales data over time to visualize month to month sales as a Key Performance Indicator (KPI). A function can be created by a user that will perform the math between the previous month and the current month that is then translated into a green or red format by Tableau, Excel, etc. automatically for a quick visual understanding for the humans consuming the information.

SCALAR, INLINE, AND MULTI-STATEMENT FUNCTIONS

Scalar, Inline, and Multi-Statement Functions are three types of UDF's, each of which have different capabilities and different use cases.

SCALAR

Scalar functions are used to return a single value. Handy for performing calculations and things given data. In the example from this week's module we see a scalar function that's created to perform simple multiplication between two numbers. Once that function is defined we SELECT the function with the two numbers in parentheses and get the result.

```
Create Function dbo.MultiplyValues(@Value1 Float, @Value2 Float)
 Returns Float
 As
  Begin
   Return(Select @Value1 * @Value2);
  End
go
-- Calling the function
Select Tempdb.dbo.MultiplyValues(4, 5);
go
```

INLINE

Inline functions return a table of results instead of a single result like a scalar function. In the example below form this week's assignment we see a function created that, when run, will return a table of data only for the KPI value want.

```
CREATE FUNCTION dbo.fProductInventoriesWithPreviousMonthCountsWithKPIs(@Value1 FLOAT)
 RETURNS TABLE
 AS
  RETURN
   (SELECT ProductName
    ,InventoryDate
    ,[Count]
    ,PreviousMonthCount
    ,CountVsPreviousMonthCountsWithKPIs
    FROM vProductInventoriesWithPreviousMonthCountsWithKPIs
   WHERE CountVsPreviousMonthCountsWithKPIs = @Value1)
;
go
```

MULTI-STATEMENT

A multi-statement function is the same as an inline function except that combines data from multiple tables or views for the results with internal joins. This can also be accomplished by doing the initial leg-work and creating a view that performs joining of data and then an inline function can perform the process. Doing it this way will increase the overall system performance since the views are cached and will not need to be rerun every time the function is performed. But a multi-statement function will still perform the work required.

Conclusion

While there are a great deal of functions that are provided by the various SQL writing programs that helps when writing code, knowing how to write a UDF will always be a good skill to have. At the end of the day the base functions are just a building block to be used when creating views and there will always be need to write our own function to serve the current business needs.