Blake Skouras

March 12, 2025

Foundations Of Databases & SQL Programming

Assignment07

Introduction to Functions

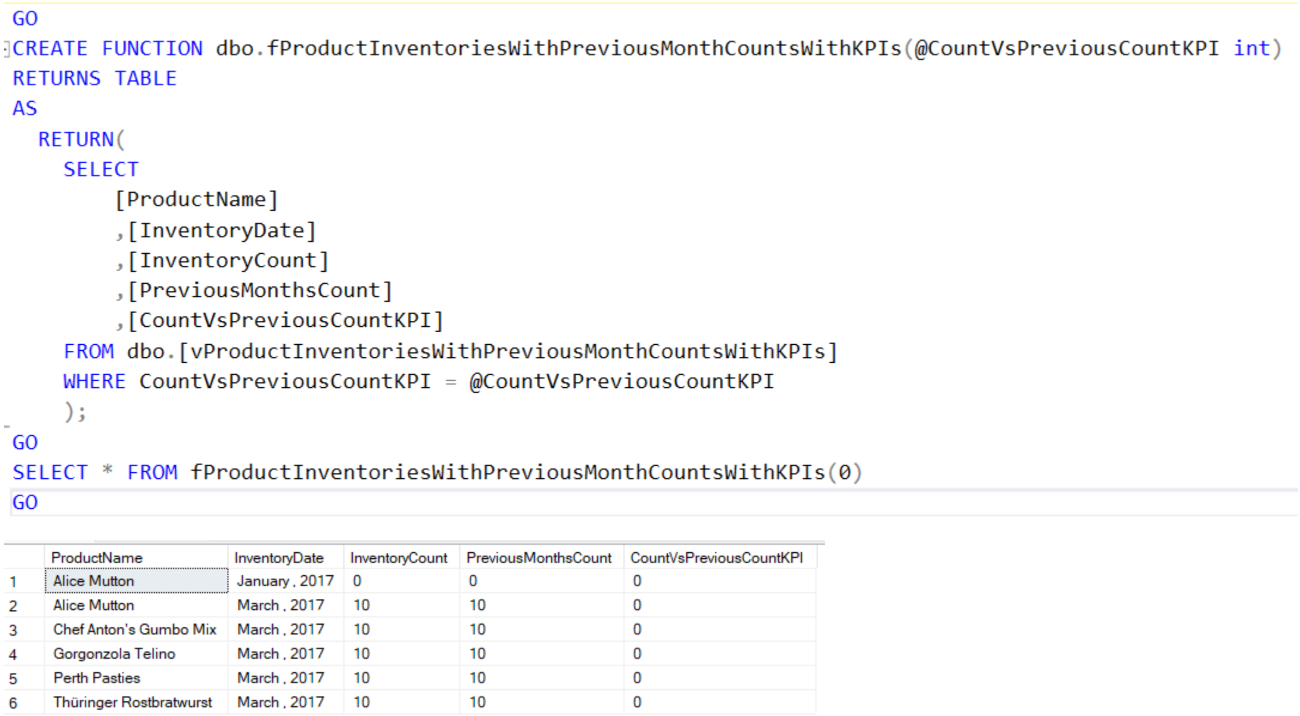
Introduction

The mission of this paper is to define some of the elements to consider when developing Functions. Each section will address one of the questions below, followed by concluding remarks.

1. ***Explain when you would use a SQL UDF.***
2. ***Explain are the differences between Scalar, Inline, and Multi-Statement Functions.***

When Would You Use a SQL UDF?

UDF stands for User Defined Function. They are a custom set of functions one can develop based on need. Like View, once written, UDFs can be "called" whenever needed. The nice thing about a UDF is it runs off of a Select Statement. It can also interact directly with Views which, when combined, provide a very efficient application to pull data. Figure 1 shows an sample UDF generated in Question 8 of the Module 7 assignment.



```
GO
CREATE FUNCTION dbo.fProductInventoriesWithPreviousMonthCountsWithKPIs(@CountVsPreviousCountKPI int)
RETURNS TABLE
AS
  RETURN(
    SELECT
        [ProductName]
        ,[InventoryDate]
        ,[InventoryCount]
        ,[PreviousMonthsCount]
        ,[CountVsPreviousCountKPI]
    FROM dbo.[vProductInventoriesWithPreviousMonthCountsWithKPIs]
    WHERE CountVsPreviousCountKPI = @CountVsPreviousCountKPI
    );
GO
SELECT * FROM fProductInventoriesWithPreviousMonthCountsWithKPIs(0)
GO
```

	ProductName	InventoryDate	InventoryCount	PreviousMonthsCount	CountVsPreviousCountKPI
1	Alice Mutton	January , 2017	0	0	0
2	Alice Mutton	March , 2017	10	10	0
3	Chef Anton's Gumbo Mix	March , 2017	10	10	0
4	Gorgonzola Telino	March , 2017	10	10	0
5	Perth Pasties	March , 2017	10	10	0
6	Thüringer Rostbratwurst	March , 2017	10	10	0

Figure 1: Sample UDF from homework

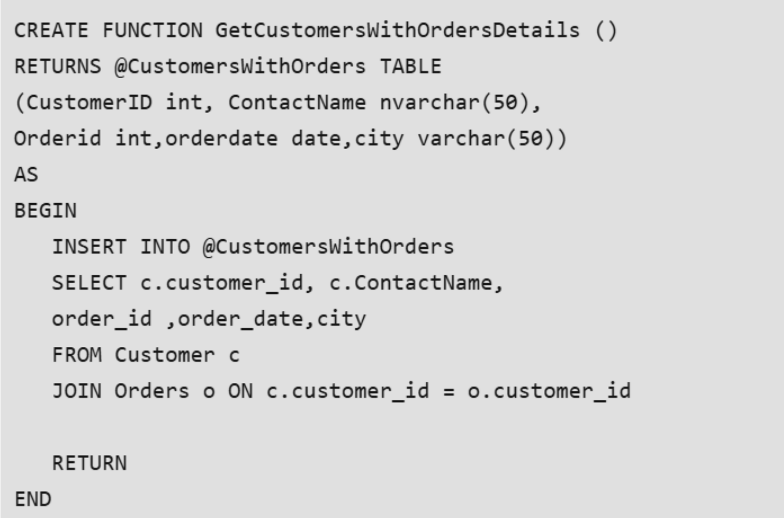
One powerful element of the UDF is the ability to define parameters for the function to work to. In the Figure 1 example the parameter (@CountVsPreviousCountKPI int) was defined. When paired with the WHERE clause "CountVsPreviousKPI = @CountVsPreviousCountKPI" dictates that if a 0, 1, or 2 are placed within the "( )" of the SELECT * FROM statement then the function will run to that value. Figure 1 shows a "0" placed in the function and the table returned only products with a KPI of 0.

The differences between Scalar, Inline, and Multi-Statement Functions

In its simplest sense a Scalar function returns a single value. The most relatable example would be a mathematic functions used in Excel (AVG, SUM, COUNT, etc). When applied to a column of numbers you can receive back one aggregated value.

Inline and Multi-Statement functions on the other hand return one or more rows and columns of data. They are also referred to as table-valued functions (TVF), meaning they return a table of data, not just one value like a Scalar function.

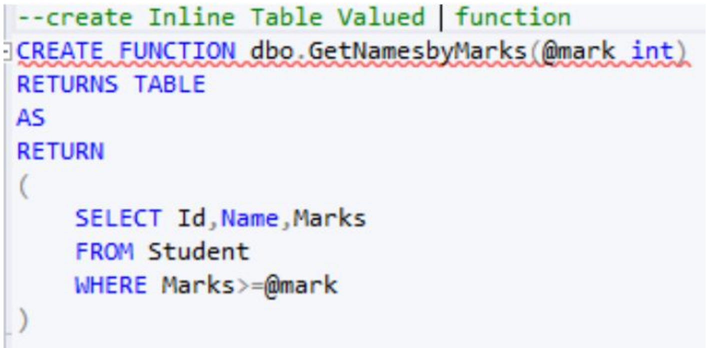
Both Inline and Multi-Statement functions utilize the “Returns Table” statement to encapsulate the Select Statement. “An Inline Table Valued Function is “inline” because its definition is part of a query, rather than being a standalone object.” (https://www.geeksforgeeks.org/inline-table-valued-function-in-sql-server/, 2025)(External Site) When written there is a notable difference between Inline and Multi-Statement functions. Following the “As” statement a Multi-statement function requires a BEGIN and END statement framing the Select statement in the query. Figure 2 provides an example of a Multi-Statement function while Figure 3 provides an example of an Inline function.



```sql
CREATE FUNCTION GetCustomersWithOrdersDetails ()
RETURNS @CustomersWithOrders TABLE
(CustomerID int, ContactName nvarchar(50),
Orderid int,orderdate date,city varchar(50))
AS
BEGIN
   INSERT INTO @CustomersWithOrders
   SELECT c.customer_id, c.ContactName,
   order_id ,order_date,city
   FROM Customer c
   JOIN Orders o ON c.customer_id = o.customer_id

   RETURN
END
```

Figure 2: Example multi-statement function (From Multi-Statement Table Valued Function in SQL Server, https://www.geeksforgeeks.org/multi-statement-table-valued-function-in-sql-server/)(External)



```sql
--create Inline Table Valued function
CREATE FUNCTION dbo.GetNamesbyMarks(@mark int)
RETURNS TABLE
AS
RETURN
(
    SELECT Id,Name,Marks
    FROM Student
    WHERE Marks>=@mark
)
```

Figure 3: Example Inline function (From Inline Table Valued Function in SQL Server, https://www.geeksforgeeks.org/inline-table-valued-function-in-sql-server/)(External)

Conclusion

Functions provide an avenue for determination of a singular value via a Scalar function or a table of values via an Inline or Multi-Statement function. The tabular output of Inline and Multi-statement functions are similar to that of a View. Once written the code can be called at any time and does not need to be imbedded in the database. Like Views, Functions can also reference Views and Tables directly making them handy for summarizing code in a more efficient manner.