Student:	Charlene Bane
Date:	November 27, 2022
Course:	IT FDN 130 A Au 22: Foundations of Databases & SQL Programming
Assignment #:	07

Assignment 07 | When to Use a SQL User-Defined Function (UDF) and
Differences between Scalar, Inline, and Multi-Statement Functions

PUT GIT HUB HERE

Introduction

The purpose of this paper is to consider the following topics:

1. *Explain when to use a SQL UDF.*

2. *Explain the differences between Scalar, Inline, and Multi-Statement Functions*

SQL User-Defined Functions (UDFs) | *When to Use*

Microsoft SQL Server has many built-in functions that can be used to manipulate data (string, numeric, date, etc.). However, the built-in functions may not meet the needs of every developer in all circumstances. Therefore, we have the option of defining our own custom functions called User-Defined Functions (or UDFs for short).

There are two types of UDFs:

- **Table-Valued Functions (TVFs):**
  TVFs return a table data type.

- **Scalar Functions:**
  Return a single data value.

According to Microsoft User-Defined Functions - SQL Server | Microsoft Learn (2022), the benefits of a user-defined function include:

- **Modular Programming:**
  A function can be created once, stored in the database, and then called as many times as needed in the code.

- **Faster Execution:**
  UDFs are cached and don't need to be reparsed and reoptimized with each use, resulting in faster execution times.

Scalar, Inline, and Multi-Statement Functions | *Differences*

Next, we will describe the differences between Scalar, Inline, and Multi-Statement User-Defined Functions*:*

*Table 1 | Scalar, Inline, and Multi-Statement Functions (*Create User-defined Functions (Database Engine) - SQL Server | Microsoft Learn *(2022)):*

	Scalar Functions	Inline Functions	Multi-Statement Functions
Description	A scalar-valued UDF accepts parameters and returns a single data value of the type defined in the RETURNS clause.	For an inline scalar function, the returned scalar value is the result of a single statement.	For a multi-statement scalar function, the function body can contain a series of Transact-SQL statements.
Differences	Returns a single value.	Returns a table of values.	Returns a table of values.
Example of a Multi-Statement Function			

```
IF OBJECT_ID (N'dbo.ufnGetInventoryStock', N'FN') IS NOT NULL
    DROP FUNCTION ufnGetInventoryStock;
GO
CREATE FUNCTION dbo.ufnGetInventoryStock(@ProductID int)
RETURNS int
AS
-- Returns the stock level for the product.
BEGIN
    DECLARE @ret int;
    SELECT @ret = SUM(p.Quantity)
    FROM Production.ProductInventory p
    WHERE p.ProductID = @ProductID
        AND p.LocationID = '6';
     IF (@ret IS NULL)
        SET @ret = 0;
    RETURN @ret;
END;
```

Conclusion

In conclusion, we defined SQL UDFs, identified the two types of UDFs, and briefly discussed the benefits of using them. Additionally, we defined and highlighted the differences between scalar, inline, and multi-statement UDFs.

This consideration highlights the agility of MS SQL Server. The developer is only limited by imagination.