

Name: Sam Dao

Date: Feb 27, 2022

Course: Foundations of databases & SQL programming

<https://github.com/daohw2022/DBFoundations-Module07>

## Assignment 07 – Functions

### Introduction

In this assignment, I've used Views, System Functions and User-Defined-Functions (UDF) in coding.

### When to use User Defined Functions

In SQL, User Defined Functions (UDF) enable developers to create functions of their own for performing certain tasks. It allows input parameters and returns either a single value (scalar) or tables. UDF is simpler to invoke from inside another SQL statement; it can be used in a Select, Where or Case statement. There are many usages of UDF. One usage is to filter data based on some complex constraints that can be expressed as an UDF. Another usage is to replace repetitive coding for frequently enacted operations to promote modular programming. Querying a returned table from UDF to join with other tables is also useful.

### Scalar, Inline, and Multi-Statement Functions

There are three different types of User Defined Functions. Each type refers to the data being returned by the function. Scalar functions return a single value. In Line Table functions return a single table variable that was created by a select statement. An example from this assignment shown in figure 1 is an inline table-valued function. The final UDF is a Multi-statement Table Function. This function returns a table variable whose structure was created by hand, similar to a Create Table statement. It is useful when complex data manipulation inside the function is required. There are some limitations for UDF; Inline table function only allows Select statement, certain operations are disallowed in UDF, such as some non-deterministic system functions (like RAND), temporary table and TRY / CATCH block.

```
CREATE OR ALTER FUNCTION dbo.fProductsByInventory()  
RETURNS TABLE  
AS  
RETURN (  
    SELECT TOP 10000 ProductName, FORMAT(InventoryDate, 'MMMM, yyyy') AS Date, Count  
    FROM vInventories AS i  
        INNER JOIN vProducts AS p  
        ON i.ProductID = p.ProductID  
    ORDER BY ProductName, InventoryDate  
);  
  
SELECT * FROM dbo.fProductsByInventory();
```

**Figure 1. Inline table-valued function**

## Summary

System Functions and User-Defined-Functions are extremely useful in SQL programming, they can greatly simplify the coding process, allow the queries to achieve complicated operations and to increase productivity.