**IT FDN 130 A - Assignment06**

**Thomas Gray**

**03-02-2021**

# **Functions**

# **Introduction**

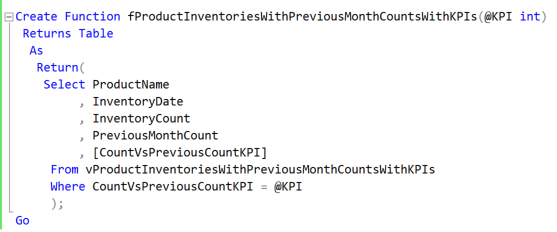
During Module seven, the class took a deeper dive into understanding, using, and creating functions. Functions are a great tool to keep in your belt when it comes to utilizing and creating databases and queries. A function can be created and ran to preform a specific set of tasks. For instance, if you are repeatedly writing the same query code over and over, you can create a function that encapsulates the code into a one-line command that can be called. This assignment will cover the questions noted below in **Figure 1,** as well as contain some SQL code submitted alongside this word document.

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

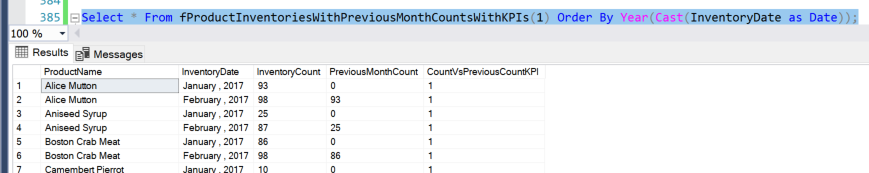
**Figure 1.**

## **Explain when you would use a SQL UDF.**

Using /creating a SQL user defined function can come in handy in multiple scenarios. Simply put, you would create a user defined function whenever you find the need to encapsulate either some advanced logic, or just a SQL query that you find yourself running quite often. Creating a user defined function will save you time in the long run by enabling you to call the function itself, rather than writing the query that is encapsulated by it. In **Figure 2** below, you will see an example of SQL code that was written to be captured in a user defined function called fProductInventoriesWithPreviousMonthCountsWithKPIs. As you can see, we create the function, use the AS statement to then write out the code we would like to be captured by this newly created function. You can see in **Figure 3** below that when you just call this function, you are returned the results of the Select statements written in **Figure 2.**



***Figure 2.***

******

***Figure 3.***

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

## **Scalar:**

A Scalar user defined function will only return a single value based off the results of the actions preformed by your function. A scalar function will return one of the following scalar data types : (int, char, varchar, etc). It cannot return an image, text, or timestamp.

## **Inline:**

An inline user defined function can be called to return a table based off of the function’s contents. The table Values that are returned will be derived from a single SELECT statement. There also is not a need to specify the table variable name or column definitions, as the results are a function of the columns referenced in the SELECT. You can not have duplicate column names due to this.

## **Multi-Statement:**

Like the Inline user defined functions, the multi-statement UDF also returns a table. It can be composed of more than one transaction SQL statement. This table also can contain Begin and End blocks, unlike the Inline table user defined functions.

## **Summary**

In summary, the use of functions can be extremely helpful and timesaving when it comes to business operations. By creating and storing these functions you not only give yourself the accessibility to these encapsulated codes, but also anyone else who may need to return similar data sets. As noted, there are multiple types of functions available to you by SQL, and there are multiple ways to create your own user defined functions. Figuring out which one you will use will all depend on what type/kind of data you are trying to retrieve.