Isaac Kareiva
IT FDN 130 (SQL)
Assignment 07
05/28/2024
[bigdawgikeboy/DBFoundations-Module07: module 7 (github.com)](github.com)

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

UDF stands for User-Defined Function, meaning it is something one needs to write out and define himself.  SQL itself has a number of built-in function (for example: sum(), cast(), GetDate(), etc.) but inevitably there will be instances where the user wants to do something specific to his project and therefore needs a UDF.  Generally, if one plans on doing multiple actions of the same type that can be generalized, a UDF is a good idea.  UDFs can also help with simplifying code so that it's more legible to read.

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

A Scalar function is one which returns a single value.  It can have multiple inputs, for example @value1, @value2, ...@valeN, but only one input; for example @value1 + @value2 + ....@valueN.

'Iinline' is Microsoft's language for what lecture describes as 'single value table' functions.  These functions return a table of results as supposed to a single (or scalar) result.  It is 'single' because it contains a single select statement (however it can still have multiple input values).  A 'Multi-Statement Function' on the other hand can have Multiple select statements.  One way this might be implemented would be within a conditional clause.  For example, if the input value is of a certain type, select statement 1 is executed, if it's of another type, select statement 2 is executed, and so forth.

Another type of function, not mentioned, are 'windowed valued' functions.  These are somewhat similar to 'group by' statements, except that, while displaying values from a column, they don't actually group by them.  Instead, 'windowed valued' functions have the over( partition by COLUMN_NAME) syntax, which performs 'group by' on a subset of the columns.  For example, if you had dating listing the heights of multiple individuals of different ages, you could group by age first, then use a windowed function to partition over min and max heights.  Doing this would mean for each age you would have a height listed, and then the min and max height aligned to that age for individuals of different heights.