Name: Taehan Kim

Date: Mar 24, 2025

Course: Foundations Of Databases & SQL Programming

**Understanding SQL UDFs and Function Types**

**Introduction**

In SQL, User Defined Functions (UDFs) provide a way for users to extend the functionality of the SQL language. UDFs allow for custom computations or operations that can be used within SQL queries, offering a powerful tool for developers and database administrators. This report will explain when to use a SQL UDF and highlight the differences between Scalar, Inline, and Multi-Statement Functions.

**When to Use a SQL UDF**

A SQL User Defined Function (UDF) is a function that allows users to define custom logic that can be executed within SQL queries. You would use a SQL UDF when you need to encapsulate a logic or calculation that is not supported natively by SQL. For example, when a specific operation needs to be reused multiple times in different queries, a UDF can be created to perform that operation. UDFs can also be used for operations like string manipulation, complex calculations, or conditional logic that cannot be easily handled by standard SQL functions.

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

**Scalar Functions**:  
A Scalar Function operates on a single value and returns a single value. It can be used anywhere a regular expression or value would be used in SQL queries (e.g., in SELECT, WHERE, or ORDER BY clauses). Scalar functions typically perform operations like string manipulation, mathematical calculations, or data conversions. An example is a function that calculates the square of a number or concatenates two strings.

**Inline Functions**:  
An Inline Function is similar to a view in SQL; it’s a function that returns a table result directly from a single SELECT statement. Inline functions are typically faster than multi-statement functions as they are compiled into the query plan by the database engine. They are more efficient than multi-statement functions, as they perform all actions in one statement, usually involving simpler operations. Inline functions return a table type and are generally used when the result of the function needs to be treated as a set of rows.

**Multi-Statement Functions**:  
A Multi-Statement Function allows for more complex logic and can contain multiple SQL statements. Unlike Inline Functions, these functions may include multiple SELECT, INSERT, or UPDATE statements. Multi-Statement Functions are typically used when the function needs to perform more complex processing, such as conditional logic, multiple operations, or updates to other database objects. However, they can be slower than Inline Functions due to the complexity and number of statements involved.

**Similarities**:  
All three function types—Scalar, Inline, and Multi-Statement—are used to encapsulate logic and can be reused in SQL queries. They all allow users to extend SQL's built-in functionality, enabling custom behavior within queries.

**Differences**:

**Scalar Functions** return a single value based on the input while **inline Functions** return a table result from a single SELECT statement. **Multi-Statement Functions** allow multiple SQL operations and are more complex, potentially returning multiple results.

**Summary**  
This report provided an explanation of SQL UDFs and discussed the situations in which they are useful, such as encapsulating custom logic and enhancing query reusability. The report also compared Scalar, Inline, and Multi-Statement functions, explaining their key differences. Scalar Functions return a single value, Inline Functions return a table result from a SELECT statement, and Multi-Statement Functions support more complex operations. Understanding these function types allows for more efficient and flexible SQL query design.