Mercedes Arteaga
May 21, 2024
Foundations of Databases & SQL Programming
Assignment 06
GitHub: https://github.com/Merarte1/DBFoundations

Views, Functions and Stored Procedures in SQL

Introduction

When working with SQL, a user may need to write different types of code. Sometimes this will be simple code, and other times it may be necessary to write complex code which can take a long time, even several days to write. In any of these cases, but especially when writing the code has taken considerable time and effort, the user may choose, as a safety precaution, to store their work where it can be found the next time it needs to be used. This can be done in a script file, or a database file in the form of a view, a function or a stored procedure. In this paper, we will briefly analyze these options and consider the similarities and differences that exist among them.

Views

A view is a simple way to store code in a database file. Views can be used in two ways:

1) Reporting Views are used for creating reports that either the same user or other individuals will be able to consult in the future.
2) Base Views are created for each table in a database and used for working with the data in replacement of the table itself. By restricting access to the table and allowing the use of the view, the user makes sure that the data is only used through this abstraction layer.

Functions

Functions are often referred to as User Defined Functions (UDFs). Functions can return tables or values. They can use parameters to change the results of the query, as follows:

- If we are using **table functions**, parameters are not valuable enough to justify the use of a function instead of a view, as similar results can be achieved by using a view with a WHERE clause.
- When using **scalar functions** to return a value, parameters are a valuable tool.

Stored Procedures

The third way to store code in a database file is through Stored Procedures. These are written in a similar way as Views and Functions, but they also have other characteristics that make them a convenient way to store complex code.

One advantage of Stored Procedures is that they are flexible in the sense that they do not need to only include one SELECT statement, but instead can include any type of statement: SELECT, UPDATE, INSERT, DELETE, or others.

Comparing Views, Functions and Stored Procedures

Up until now, we have described Views, Functions and Stored Procedures. In this section, we will list the similarities and differences that we can find among these 3 tools:

- The syntax to write Views, Functions and Stored Procedures is similar, but the one for Functions is slightly more complicated. It can be said that, in this sense, Views and Stored Procedures are more similar to each other than to Functions.
- Unlike Functions and Stored Procedures, Base Views allow for an individual to restrict access to a table to the public or a specific set of people and grant access to the view created for that table, in this way forcing the use of an abstract layer instead of the original table.
- Unlike views, Functions can be used to return a table or a scalar value, and they can use parameters to change the results of the query. However, table functions do not have an advantage over views, as views can be used with a WHERE clause and achieve similar results.
- Views and functions can be consulted with a SELECT keyword, whereas Stored Procedures require that the code be executed. At the same time, consulting functions require that we use the “dbo” prefix and a set of parentheses at the end.
- Stored Procedures allow for the use of any statement, not only the SELECT statement like Views and Functions. In this way, this tool is more flexible and convenient for storing complex code.

Summary

In conclusion, the tools described, Views, Functions and Stored Procedures are useful ways to save a particular written code. Depending on the purpose of the query, and perhaps even the preference of the management of the company where the user is working, any of these options may be implemented. There are several characteristics that distinguish them from one another: the way the code is written, their flexibility and their execution.