

Spencer Neiffer

11/18/24

Assignment06 WriteUp

ITFDN130A

Introduction

In this document, I will explain the scenario in which SQL Views should be utilized. I'll also discuss the similarities and differences between three important SQL tools: Views, Functions, and Stored Procedures. These tools are crucial to data integrity and database efficiency.

SQL Views

A view in SQL is essentially a saved SELECT statement that extracts data from the database to a virtual table. That virtual table (view) can be queried and edited just as the real table can be, but because it doesn't store the actual data, there is no risk to the main table. Its main purpose is to serve as an Abstraction Layer to the actual tables and raw data in the database. Using views as an abstraction layer gives users an interface with which to interact with the database, instead of interacting with the actual tables and data within it.

Views can improve data integrity and security within a database. For instance, views can be protected from unwanted changes using SCHEMABINDING. If a change were to be made to the table that the view is based on and that change would affect the view, the change to the table would be blocked, thus protecting the integrity and consistency of the view. In addition, views can be restricted to specific user groups, making it impossible for the specified group to access the entire view or certain data within it, which improves data security.

One of the most important uses of a view is to represent the tables within a database. This means that for each table, a view should be created to represent it, and any future queries should use the views instead of the tables. Views should also be created for reporting purposes. By saving a select statement in a view, reporting data can easily be retrieved from the database by using the view instead of writing a new query. This is especially useful when working with complex queries. Views can also be used to hide data by restricting access to views to specific user groups,

increasing security. Another use case for views is to simplify database interactions for users. Complex tables and data can be hidden behind views, making the database easier to use.

Views vs Functions vs Stored Procedures

Views differ from functions and stored procedures in that they're essentially virtual tables that represent actual tables, and pre-written select queries that retrieve data from one or tables or views. Stored procedures are similar to views and functions in that they are pre-defined SQL code. However, they contain the code in pre-compiled blocks and are capable of insert, update, and even delete operations. Like views and stored procedures, functions are pre-written and reusable, but they only calculate and return a single value based on specified parameters. Both views and stored procedures are SQL objects that either change the database or how the data is viewed, where functions don't directly change data but return a result based on it.

Summary

In closing, SQL views have many uses that include: acting as an abstraction layer, restricting access to specific data, simplifying interaction with the database, and running reports. Views present data from tables and can change how that data is seen, without changing the actual tables. Stored procedures can manipulate actual tables and data, but don't necessarily return results. Functions return a result but don't manipulate data. All three tools can be created and stored for later use.

Citations

<https://www.geeksforgeeks.org/sql-views/>

<https://learn.microsoft.com/en-us/sql/t-sql/statements/create-function-transact-sql?view=sql-server-ver16>