Name: JohnRobert Wilson
Date: November 12, 2023
Course: IT Foundations of Database Management

Assignment 06 - Views

Introduction

In Module 6 we worked with views to construct “reports” that can be pulled with a simple SELECT * statement. In addition we touched on functions, stored procedures, and using Github to store/share projects.

Explain when you would use a SQL View.

Views and intermediary space that provides flexibility and security to your tables and data. They are stored within the database directly, and can be called like a table, this allows you and your user to access queries with a simple select statement even if the foundation is rather complex. This also allows data to be refactored allowing some ways to change a database without breaking an end-users tools For example, if you had a table with first and last name combined you could rebuild the database with them separate, and then create a view where that column is concatenated using the same name as the original database. Lastly by using views you can grant user access to certain data through the views, while preventing access to the initial tables.

Explain are the differences and similarities between a View, Function, and Stored Procedure.

Functions are similar to views since they allow you to return tables or results based on a SELECT statement. The advantage or a function over a view is that it can accept parameters into the function, whereas the view would apply the parameters into the where statement of the subsequent Select statement of the view. Stored procedures go farther allowing operations beyond the select statements.

Appendix

SQL code saved as a file on a HD/Cloud is called a script. To store a code in your database you need to create views, functions, or stored procedures . In Module 06 we will explore all of these.

This DB stored statements are stored in a table that can be viewed by using the following commands

- Select * FROM SysComments - Pulls as machine code
- SP_helpText [View name] - Pulls as the action statement with original formatting.

Views

- Allow you to store a select statement. It creates a view with that select statement.
- Order by class shouldn't be used in the view. Ideally you would apply the order by on the Select View statement. Think of the view as the command on what to be pulled… but it isn't for the actually manipulation of the data.
- **Horizontal partitioning:** You can filter rows in a view statement by using the where clause. → it is a classic select statement that is stored. NO changes have been made to the database, or data being stored in this format… it is ONLY a select statements.
- WITH SHCEMABINDING will prevent a table referenced by a view from being dropped or having a column changed. Any change that would break a view will end in error.
-

- **Vertical Partition:** UseCase: you only want users with certain permission to see certain data. (HR vs phonetool)
- Views must be created on a database where you have permission to do this.
- TABLE BASE VIEW: Best practice to automatically create a base view of a database to allow for changes to the tables that don't break the database.

Functions AKA User Defined Functions (UDF)

- Functions will require dbo. And in the select statement of a function it will need the () at the end. This will work with direct inputs… but the sub-query needs to be set as an separate clause.