

Name: Sam Dao

Date: Feb 19, 2022

Course: Foundations of databases & SQL programming

<https://github.com/daohw2022/DBFoundations>

Assignment 06 – Views

Introduction

In this assignment, I've learned Views, Functions and Stored Procedures for SQL. The assignment includes using View, Schemabinding, Deny and Grant access to create Base View and other customized Views.

When to use Views

View is a SQL statement (query) that is stored in the database with an associated name. Unless indexed, a view does not exist as a stored set of data values in a database. The rows and columns of data coming from tables referenced in the query are produced dynamically when the view is referenced.

A view can be created from one or many tables or from other views in the current or other databases. It acts as a filter on the underlying tables referenced in the view. View can be used to customize data appearance or to summarize data from various tables for general reports. It can limit the degree of exposure of the underlying tables to users by creating a subset of the data contained in a table, for instance, a given user may have permission to query the view, while being denied access to the rest of the base table. Finally, a view can provide a backward compatible interface to emulate a table that used to exist but whose schema has changed. In figure 1, a view, 'vInventoriesByEmployeesByDates', is created by retrieving some data from two tables. The displayed employee's name combines two separated fields of firstname and lastname to appear more natural in a report. In addition, without access permission to the original tables, other data are shielded.

```
CREATE VIEW vInventoriesByEmployeesByDates
AS
SELECT TOP 10000 InventoryDate, Employee = EmployeeFirstName + ' ' + EmployeeLastName
FROM dbo.vInventories AS i
    INNER JOIN dbo.vEmployees AS e
    ON i.EmployeeID = e.EmployeeID
GROUP BY InventoryDate, EmployeeFirstName, EmployeeLastName
ORDER BY InventoryDate;
GO
```

Figure 1. SQL coding to create view

Views, Stored Procedures and User Defined Functions

Views, Stored Procedures and User Defined Functions (UDF) are database objects, which contain a set of SQL statements (queries) to complete a task. Oftentimes a developer will create views, functions or stored procedures to simplify the coding process for repeating run queries.

While View can contain only one single Select query, a Stored Procedure is an encapsulation of SQL statements that can be called from an application (or run in SQL Management Studio). Stored Procedure is often used to perform Insert, Delete, Update statements on data as well as loops, and calls to other stored procedures, tables and views. Figure 2 illustrates the Transaction and Try-Catch block statements wrapped within a stored procedure.

Unlike View, Stored Procedure can perform modification to tables, accept parameters and return values. In addition, Stored Procedure is compiled for the first time and its compiled object is saved, which executes when called upon using the Execute statement. In contrast, Views and Functions have to be compiled and optimized every time they are executed.

```
CREATE PROCEDURE DeleteCondiments @CategoryName NVARCHAR(100)
AS
BEGIN

BEGIN TRY
    BEGIN TRANSACTION;
    DELETE FROM Inventories
    WHERE Inventories.ProductID IN
        (SELECT ProductID
         FROM Products
         WHERE Products.CategoryID IN
            (SELECT CategoryID
             FROM Categories
             WHERE CategoryName = @CategoryName));

    DELETE FROM Products
    WHERE Products.CategoryID IN
        (SELECT CategoryID
         FROM Categories
         WHERE CategoryName = @CategoryName);

    DELETE FROM Categories
    WHERE CategoryName = @CategoryName;
    COMMIT TRANSACTION;
END TRY

BEGIN CATCH
    PRINT Error_Message ();
    IF @@TRANCOUNT > 0 ROLLBACK TRANSACTION;
END CATCH;
```

```
END
```

```
GO
```

```
EXEC DeleteCondiments @CategoryName = 'Condiments';
```

```
GO
```

Figure 2. Application of a stored procedure from assignment 04

A User Defined Function (UDF) is a special type of procedure that can return a single value (scalar) or a table while View only works with tables. UDF can be used in the Where/Having/Select section of the SQL statement. It can be useful for calculation and return a single value for comparison or for checking. However, Function allows only input parameters but not output parameters.

Summary

View acts as a data abstraction process of hiding unwanted details from the end user. It helps in achieving data independence which is used to enhance the security of data. User Defined Functions (UDF) are useful for frequent and repeatedly executed queries, supporting the concept of modular programming. Another usage is to filter data based on some complex constraints that can be expressed as an UDF. Stored procedures are compiled once and stored in executable form, so procedure calls are quick and efficient. Executable code is automatically cached and shared among users. This lowers memory requirements and invocation overhead. By designing applications around a common set of stored procedures, developers can avoid redundant coding and increase productivity.