Marci DeLeon

November 21, 2021

IT FDN 130: Foundations of Databases and SQL Programming

Module 6

<https://github.com/Marci-student/DBFoundations>

An Essay With a View

# Introduction

Views are one of the single most important aspects of SQL. Views allow people with little concept of coding or database structure or rules to access the information of a database, without fear of the database itself becoming corrupted. In this essay, I will describe when a View should be used, and the differences and similarities between a View, a Function, and a Stored Procedure.

# When Should a View Be Used?

Unless someone is making changes directly to a database (e.g., creating one, modifying the data, or deleting one), you should always be using a View. A View allows for the data to be transformed, analyzed, or ties into little bows without the underlying data itself ever being touched. This means that the same data can be analyzed in many different ways, and that many different people can have access to that analysis, but that the data is never actually changed. The View puts a “layer of abstraction” between the user and the data.

For example, the FDA works with private companies when an FDA-regulated product is found to be potentially injurious to human or animal health. FDA’s recall coordinators work with companies to remove those products from the market. In their work, these recall coordinators must document the associated activities; with the Government Paperwork Elimination Act, that documentation has been moved to electronic storage, in a database.

Permission-based access allows many people to look at this information, but only allow certain people to add or modify records. This ensures that the information that goes into the database is controlled and that the people who change it are trained in how to perform this activity.

However, many people who are not recall coordinators need to be able to access this data and analyze it in different ways. FDA’s field data warehouse takes all of this recall information and creates views which allow people both skilled and unskilled in SQL and the FDA’s specific databases to get important data.

For example, there are reports that experienced SQL programmers have created and validated which allow the same type of information to be pulled, and all a user has to do is put in the timeframe and programs that they wish to see. This is a good example of a Reporting View – one person (or team) created the report, and then saved it for anyone to use.

However, if there is not a Reporting View available for the information someone wants, they can then turn to people who are skilled in the use of the database, if not in SQL. These people can use the graphical interface that was created to allow people who did not know SQL to still interact with the code, allowing users of all types to interact with the Basic Views that were created for each of the databases within the data warehouse. Anyone can create their own Reporting Views off of the Basic Views, and neither the Basic Views nor the underlying tables are affected in any way.

This allows for the same information to be used in many different ways throughout the FDA, and still keep the integrity of the painstakingly gathered data.

Views should be used in most circumstances that do not require direct manipulation of the database.

# Views, Functions, and Stored Procedures

## What is a View?

A view is a SQL select statement (simple or complex) which returns a table. This table can then be manipulated by a user in the same way that the tables in a database could be, except that the views don’t affect the underlying data.

## What is a Function?

A function is created with similar coding to a View, but it is meant to manipulate the data in some way. This manipulation can result in either a table, or a scalar value.

It is generally a complicated *select statement* which can be saved to be reused in many different statements. For example, SQL understands a calendar year with calendar months. But what if you need a timeframe within a fiscal year, or a contract year, or a religious year. In these cases, January 1-December 30 is a poor timeframe to work with. If your fiscal year goes from July 1-June 30 (as it does for Washington State government), you may create a function which takes any given date and provides the correct fiscal year. Since this is an annoying bit of code to have to write for every single query you have, it’s worthwhile to have this function and put it into any query that you have for which you want to group things by the fiscal year.

A function must be bracketed by “Begin” and “End” statements.

## What is a Stored Procedure?

A stored procedure has similar coding to a view or a function. In fact, just as every square is a rectangle, but not every rectangle is a square, so too is every function a stored procedure, but not every stored procedure is a function. This is because a stored procedure is more complex than a function: a function can only have a select statement, while a stored procedure can use any sort of coding available. Also, a function can be part of a stored procedure, while a stored procedure cannot be part of a function.

While it may seem like a stored procedure should always be used in place of a function, functions are very good at being re-used. For the most part, a stored procedure has to run and output must be acted upon, while a function can be simply added into a select statement.

## Summary

Views are phenomenally import parts of SQL; they allow databases to be used while keeping the data itself out of the way of any changes or manipulations. Views are abstracted tables, and functions and stored procedures can be used in them (or directly in tables) to manipulate, change, or otherwise work with the data in them.