##### **Clariecia Groves**

##### **August 17, 2020**

##### **IT FDN 130 A - Foundations of Database Management**

##### **Assignment 06**

<https://github.com/CGroves3-UW/DBFoundations>

**Creating Views in a Database**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

# Introduction

For this assignment, I learned more about how to create databases, joins, views, functions, and stored procedures. Joins are helpful for extracting data from multiple tables. Views, functions, and stored procedures allow you to save simple and complex select statements so that they can be stored into a database. These options provide an easy way to execute saved code using simple select statements which makes it easier for business users to access various reports and data.

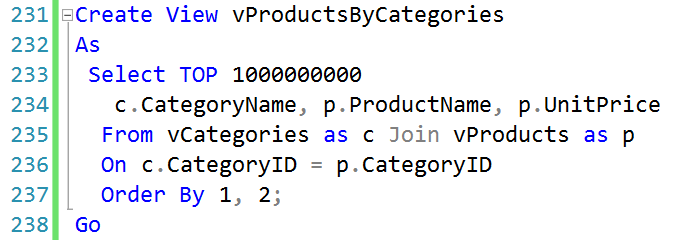
# SQL View

Explain when you would use a SQL View.

SQL Views contain rows and columns just like a real table except they “display data” as opposed to “store data”. As such, SQL Views are used when you need to frequently access a specific view of the data. SQL Views are also used for:

* Providing easy access to structured data.
* Setting up the data in a desired format for end users.
* Restricting access to the data by denying access to tables while granting access to views.
* Generating reports that summarize data from multiple tables.

In this assignment we had separate tables for products and categories. The products table contained the ProductID, ProductName, CategoryID, and UnitPrice. The categories table contained the CategoryID and CategoryName. To create a view that shows a list of category and product names along with the unit price of each product, you would have to create a view that includes statements that join the two tables on CategoryID as shown in Figure 1.



**Figure 1: SQL View with Join on the Categories and Products Tables**

# Views, Functions, and Stored Procedures

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

In this section, we will highlight some of the differences and similarities between Views, Functions, and Stored Procedures.

* A View is a virtual table based on the result-set of an SQL statement.
* A Function is a user-defined function that accepts parameters, performs complex calculations, and returns values.
* A Stored Procedure is a subroutine of SQL statements that can be saved so that the code can be used multiple times.

Figure 2 provides a table comparison of some of the similarities and differences between views, functions, and stored procedures.

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Views | Functions | Stored Procedures |
| Accepts Parameters | No | Yes | Yes |
| Restricted to Select Statements | Yes | Yes | No |
| Allows modifications to tables | No | No | Yes |

**Figure 2: Similarities and Differences Between Views, Functions, and Stored Procedures**

# Summary

After completing this module which included reading supplemental websites, watching videos, and practicing exercises, I was able to successfully create views in a database. This written assignment demonstrates my knowledge of joining tables using inner and self-joins along with creating views. I am excited to continue learning more about databases in future modules and hope to build my own database in practice.