Name: Lauren Sprung

Date: 11/22/2022

Course: Foundations of Databases & SQL Programming

Assignment 6 – Views

Introduction

In this week's module, I learned about the use of views, functions, and stored procedures to protect the data in a database and make data reporting more efficient. I also practiced creating views from the select statements that I created in last week's assignment. I set permissions on the views and tables to protect the data in the tables.

SQL View Scenarios

Views are useful in several scenarios. While there are some situations that allow a user to alter the table through a view, typically a view is used for querying data in a database. Therefore, a view can be used as separation between the user and the table. A type of view which queries for all the data in one table is called a "base view". The purpose of the base view is to allow a user to query the data in a table without having to interact directly with the table. Figure 1.1 below shows the SQL code to deny a user privileges to a table but grant permissions to the base view of that table.

```sql
Deny Select On LaurensTable to Public;
Grant Select On vLaurensTable to Public; --the Base view of LaurensTable
```

Figure 1.1 SQL code to deny a user privileges to a table and grant privileges to the base view

Another important purpose of views is to save a SQL Select statement. If a SQL statement is complex and will be useful more than once, a user can save that Select statement in a "Reporting" view. Just like in other Select statements, a view can query data from more than one table. One special consideration is that in order to use the Order By clause in a view, the query would need to select the Top X amount of data. The number X would need to be sufficiently large to account for all the rows in the table. Figure 1.2 shows the creation of a view with an Order By clause.

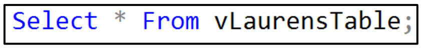
```sql
Create View [dbo].[vLaurensTable]
As
    Select Top 10000000 LaurensFriends, LaurensFriendsAddresses
    From LaurensTable
    Order By LaurensFriends, LaurensFriendsAddresses;
```

Figure 1.2 SQL code to create a View with an Order By clause

Lastly, a view can be used as an abstraction layer. If an application queries the data stored in a database, it is helpful to link the application to a view rather than to a table directly. In this case, if there are required changes to the way the data is stored, it won't necessarily affect the view.

Compare Views, Functions, and Stored Procedures

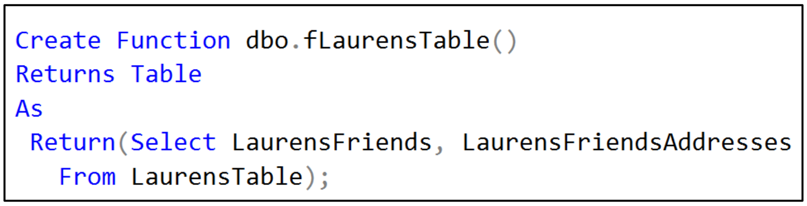
As discussed above, a view is a saved query. The results of a view are the column or set of columns called out inside the query. Views will always represent the latest data in the table from when the view is executed because it does not actually store any data. Figure 2.1 shows the code to query the data from a view. The results of a view can only be used as part of the From clause in a Select statement.



```sql
Select * From vLaurensTable;
```

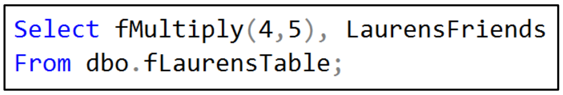
Figure 2.1 SQL code to query the data in a view

A function is like a view in that it can return a table, but it can also return a scalar. This feature enables it to be used in the select clause of the select statement. Another difference between views and functions is that functions can take parameters. The parameters can change the results of the query. Figure 2.2 shows the code to create a function with parameters. Figure 2.3 shows the use of functions in the select and from clauses of select statement.



```sql
Create Function dbo.fLaurensTable()
Returns Table
As
 Return(Select LaurensFriends, LaurensFriendsAddresses
   From LaurensTable);
```

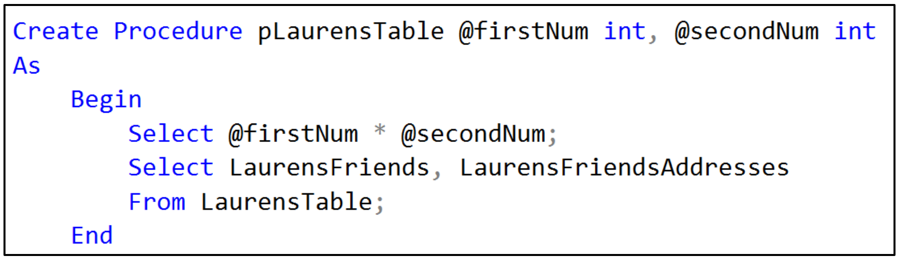
Figure 2.2 SQL code to create a Function



```sql
Select fMultiply(4,5), LaurensFriends
From dbo.fLaurensTable;
```

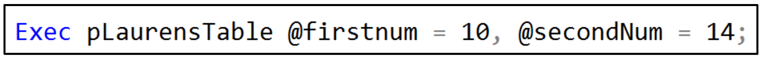
Figure 2.3 A select statement using the results of a function in the select and from clauses

Lastly, a stored procedure is a saved set of SQL statements. Unlike views, a stored procedure can have several SQL statements. Like functions, the stored procedure can accept parameters. Figure 2.4 shows the SQL code to create a stored procedure. Another difference between stored procedures vs views and functions is how they are called. Unlike views and functions, a stored procedure cannot be used in a Select statement. Figure 2.5 shows the code to execute a stored procedure.



```sql
Create Procedure pLaurensTable @firstNum int, @secondNum int
As
    Begin
        Select @firstNum * @secondNum;
        Select LaurensFriends, LaurensFriendsAddresses
        From LaurensTable;
    End
```

Figure 2.4 SQL code to create a Stored Procedure



```sql
Exec pLaurensTable @firstnum = 10, @secondNum = 14;
```

Figure 2.5 SQL code to execute a Stored Procedure

Summary

In this module, I learned about Views, Functions, and Stored Procedures. I learned about scenarios where you would use each of them. I practiced creating base views and views from more complex statements. I look forward to next week to dive deeper into functions.