Julia York
15-NOV-2025
Foundations of Databases and SQL Programming
Assignment 06
https://github.com/juliayork2/DBFoundations

Views, Functions, and Stored Procedures in SQL

Introduction

Views are used in SQL to create an abstraction layer which allows users to interact with the database without impacting the underlying tables. Views are used both to prevent unintended changes and to create a consistent experience for users. Functions and stored procedures are similar to views in that they are named saved SELECT statements. However, views, functions, and stored procedures each serve a different purpose within SQL.

SQL Views

In SQL, a view is a saved SELECT statement. When a view is accessed, the saved SELECT statement is executed using the data from the table referenced in the SELECT statement. However, actions taken while using the view will not be reflected in the referenced table. In this way, a view creates an abstraction layer that allows users to interact with the database without changing the data contained within the database or the underlying structure of the database. Abstraction layers serve both to protect the database from unintended changes and to create a consistent experience for users. The table referenced in a view may be altered, but the table view can be formatted to preserve the original presentation of the data. Views can also be used to limit access to sensitive information by allowing users to access only some of the data contained within a table. For example, in Figure 2 a view was created to display only the LastName, FirstName, and Title columns from the Microsoft Northwind Employees table (Figure 1). The command CREATE VIEW is used with a SELECT statement to create a view (Figure 2). The view can then be selected against to retrieve data.

	EmployeeID	LastName	FirstName	Title	TitleOfCourtesy	BirthDate	HireDate	Address	City
1	1	Davolio	Nancy	Sales Representative	Ms.	1948-12-08 00:00:00.000	1992-05-01 00:00:00.000	507 - 20th Ave. E. Apt. 2A	Seattle
2	2	Fuller	Andrew	Vice President, Sales	Dr.	1952-02-19 00:00:00.000	1992-08-14 00:00:00.000	908 W. Capital Way	Tacoma
3	3	Leverling	Janet	Sales Representative	Ms.	1963-08-30 00:00:00.000	1992-04-01 00:00:00.000	722 Moss Bay Blvd.	Kirkland
4	4	Peacock	Margaret	Sales Representative	Mrs.	1937-09-19 00:00:00.000	1993-05-03 00:00:00.000	4110 Old Redmond Rd.	Redmond
5	5	Buchanan	Steven	Sales Manager	Mr.	1955-03-04 00:00:00.000	1993-10-17 00:00:00.000	14 Garrett Hill	London
6	6	Suyama	Michael	Sales Representative	Mr.	1963-07-02 00:00:00.000	1993-10-17 00:00:00.000	Coventry House Miner Rd.	London
7	7	King	Robert	Sales Representative	Mr.	1960-05-29 00:00:00.000	1994-01-02 00:00:00.000	Edgeham Hollow Winchester Way	London
8	8	Callahan	Laura	Inside Sales Coordinator	Ms.	1958-01-09 00:00:00.000	1994-03-05 00:00:00.000	4726 - 11th Ave. N.E.	Seattle
9	9	Dodsworth	Anne	Sales Representative	Ms.	1966-01-27 00:00:00.000	1994-11-15 00:00:00.000	7 Houndstooth Rd.	London

Figure 1. Nine columns from the Employees table in the Microsoft Northwind database. Sensitive data, including employee dates of birth and home addresses, are present in the table.

```
Create View vEmployees
 As
    Select LastName, FirstName, Title
    From Northwind.dbo.Employees;
Go
Select * From vEmployees;
Go
```

	LastName	FirstName	Title
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Sales Representative
4	Peacock	Margaret	Sales Representative
5	Buchanan	Steven	Sales Manager
6	Suyama	Michael	Sales Representative
7	King	Robert	Sales Representative
8	Callahan	Laura	Inside Sales Coordinator
9	Dodsworth	Anne	Sales Representative

Figure 2. SQL code (top) and results (bottom) from a view created to display the LastName, FirstName, and Title columns from the Microsoft Northwind Employees table. This view could be used to limit access to sensitive information.

Differences and Similarities Between Views, Functions, and Stored Procedures

Views, functions, and stored procedures in SQL are similar in that each can be used to save a named SELECT statement in a database. Views, functions, and stored procedures can all increase efficiency by allowing frequently used SELECT statements to be saved and used repeatedly without needing to re-write code. However, there are differences in the purpose, capabilities, and syntax of views, functions, and stored procedures.

As described above, views are used primarily to create an abstraction layer that allows users to retrieve data and perform queries without affecting the underlying database. Functions are used to compute and manipulate a value without modifying the database. Functions include the built-in functions within SQL Server, such as sum and average, and user defined functions (UDFs) which are created by the user. Functions can be written to return results as a table or to return a single (or scalar) value. Functions differ from views because functions contain parameters. Views cannot contain parameters; however, the WHERE clause can be used to achieve the same result within a view as a parameter within a function. The command CREATE FUNCTION is used to create a function (Figure 3). Unlike views and stored procedures, the SELECT statement in a function is written within a RETURN statement. The RETURNS keyword is used to specify whether the results will be returned as a table or a scalar value. After creating a function, a SELECT statement can be used to view the results (Figure 3).

```
Create Function dbo.fEmployees()
 Returns Table
 AS
   Return(
   Select LastName, FirstName, Title
   From Northwind.dbo.Employees
    );
Go
Select * from dbo.fEmployees();
Go
```

	LastName	FirstName	Title
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Sales Representative
4	Peacock	Margaret	Sales Representative
5	Buchanan	Steven	Sales Manager
6	Suyama	Michael	Sales Representative
7	King	Robert	Sales Representative
8	Callahan	Laura	Inside Sales Coordinator
9	Dodsworth	Anne	Sales Representative

Figure 3. SQL code (top) and results (bottom) from a function created to display the LastName, FirstName, and Title columns from the Microsoft Northwind Employees table.

Stored procedures are used primarily to store complex logic and to perform a sequence of operations which may include modifying the database or returning multiple values. Unlike views and functions, stored procedures are not restricted to SELECT statements. Stored procedures may contain INSERT, UPDATE, or DELETE statements, which allow stored procedures to modify the underlying database. Like functions, stored procedures may contain parameters. However, unlike functions, stored procedures may return a single value, multiple values, or no values. Stored procedures tend to be used for more complex applications than views or functions, which are more limited in scope than stored procedures. The command CREATE PROCEDURE is used to generate a stored procedure (Figure 4). Stored procedures cannot be selected against. Instead, the EXECUTE command is used to run a stored procedure (Figure 4).

```
Create Procedure pEmployees
 AS
   Select LastName, FirstName, Title
   From Northwind.dbo.Employees;
Go
Execute pEmployees;
Go
```

	LastName	FirstName	Title
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Sales Representative
4	Peacock	Margaret	Sales Representative
5	Buchanan	Steven	Sales Manager
6	Suyama	Michael	Sales Representative
7	King	Robert	Sales Representative
8	Callahan	Laura	Inside Sales Coordinator
9	Dodsworth	Anne	Sales Representative

Figure 4. SQL code (top) and results (bottom) from a stored procedure created to display the LastName, FirstName, and Title columns from the Microsoft Northwind Employees table.

As shown in Figures 2-4, views, functions, and stored procedures can be used to generate the same results in some instances. It is important to understand the different uses of views, functions, and stored procedures in order to select the most suitable option for any given task. The similarities and differences between views, functions, and stored procedures are summarized in Figure 5.

	View	Function	Stored Procedure
Primary Purpose	• Retrieve data • Create abstraction layer	• Compute values	• Store complex logic • Perform sequences of operations
Allows Parameters?	No	Yes	Yes
Can Modify Database?	No	No	Yes
Syntax	CREATE VIEW *View Name* AS SELECT *Column1, Column2, ...* FROM *Table Name*;	CREATE FUNCTION *Schema Name.Function Name*() RETURNS *[Data Type]* AS RETURN(SELECT *Column1, Column2, ...* FROM *Table Name*);	CREATE PROCEDURE *Procedure Name* AS SELECT *Column1, Column2, ...* FROM *Table Name*;
Command Used to Return Results	SELECT	SELECT	EXECUTE

Figure 5. Key similarities and differences between views, functions, and stored procedures in SQL.

Summary

A view is a saved SELECT statement in SQL Server that can be queried in the same way as a table without changing the underlying data being referenced in the view. Views are used to protect data and to create a consistent user experience by acting as an abstraction layer between the user and the underlying data. Functions and stored procedures are similar to views as all are named saved SELECT statements. Functions differ from views because functions contain parameters that are used to compute a value without changing the database. Stored procedures may also contain parameters, as well as INSERT, UPDATE, and DELETE statements. As such, stored procedures may be used to modify the underlying database, which is not true of views or functions. While similar, views, functions, and stored procedures each serve a unique purpose within a database.