Name: Yuliya Kozar
Date:03/05/2025
Course: Foundations of Databases & SQL Programming
https://github.com/Y-Kozar/DBFoundations

Assignment 06

Introduction

In Assignment 6, we will explore the purpose and appropriate use of the *SQL VIEW* statement and examine and highlight the similarities and differences between *VIEW, FUNCTION*, and *STORED PROCEDURE* statements.

Explaining when to use a SQL View

A *VIEW* is a virtual table whose contents are defined by a query. A SQL *VIEW* is useful for simplifying complex queries without rewriting the logic each time. It can also improve data security by allowing users to access data through the view, without granting direct access to the underlying tables of the query. Additionally, *VIEW* statement can make your database more user-friendly by providing a simplified interface for common queries.

Explaining the differences and similarities between a View, Function, and Stored Procedure

A *VIEW*, *FUNCTION*, and *STORED PROCEDURE* are all SQL objects used to improve database efficiency, though each has distinct functionalities. A *VIEW* is similar to a table and consists of a set of named columns and rows. It is a virtual table that returns a specific, pre-defined query result without modifying data. A *FUNCTION*, on the other hand, is a block of code that accepts parameters, performs calculations or operations, and always returns a single value or a table of values. Additionally, just like the *VIEW* statement, a *FUNCTION* cannot perform permanent database changes. In contrast, a *STORED PROCEDURE* is a set of SQL statements that can perform complex operations, such as updating and deleting, while modifying data in the database. Long story short, while *VIEWS* are designed to simplify data presentation, *FUNCTIONS* are used for calculations and transformations, and *STORED PROCEDURES* manage complex data manipulation, making permanent data changes to the database.

Summary

In this paper, we investigated the purpose and proper application of the SQL *VIEWS* statement. We compared and contrasted SQL *VIEW* with *FUNCTIONS* and *STORED PROCEDURES*, examining their similarities and differences to gain a better understanding of their unique roles in database management.