What is difference between function , view and Stored procedure ?

Stored procedures, functions, and views are database objects in SQL Server, but they serve different purposes and have distinct characteristics

1. **Stored Procedure:**

- A stored procedure is a set of SQL statements that can be executed as a single unit.
- It can contain input and output parameters, allowing it to accept values and return results.
- It can include control-of-flow language elements such as conditionals and loops.
- It is primarily used for performing an action, such as modifying data or executing a sequence of SQL statements.
- It does not return a result set directly but can modify data and produce output parameters.

Example:

```
CREATE PROCEDURE GetEmployeeCount
      AS
BEGIN
   SELECT COUNT(*) AS EmployeeCount FROM Employees
END
```

2. **Function:**

- A function is a set of SQL statements that return a single value or a table.
- It can be used in SQL queries, expressions, or as part of other functions and procedures.
- Functions can be categorized as scalar functions (returning a single value) or table-valued functions (returning a table).
- It cannot modify data; its primary purpose is to encapsulate logic and return a specific value.

Example:

1.scalar function :

```
CREATE FUNCTION CalculateCircleArea(@Radius FLOAT)
RETURNS FLOAT
AS
BEGIN
    RETURN PI() * POWER(@Radius, 2);
END;
```

2.table-valued function :

```
CREATE FUNCTION GetEmployeesByDepartment(@DepartmentID INT)
RETURNS TABLE
AS
RETURN (
    SELECT * FROM Employees WHERE DepartmentID = @DepartmentID)
```

3. **View:**

- A view is a virtual table based on the result of a SELECT query.
- It does not store data itself but provides a way to represent the result of a query as if it were a table.
- Views can be used to simplify complex queries, enforce security by limiting access to specific columns, and encapsulate logic. →by using alise name
- They are read-only and do not allow modification of underlying data directly.

Example:

```
CREATE VIEW EmployeeDetails AS
    SELECT EmployeeID, FirstName, LastName, DepartmentName
     FROM Employees JOIN Departments
     ON Employees.DepartmentID = Departments.DepartmentID;
```

In summary:

- Stored procedures are used for performing actions, often involving the modification of data.
- Functions are used to encapsulate logic and return a value or a table.
- Views are virtual tables based on the result of a SELECT query, providing a simplified way to interact with the data.

Function vs stored procedure

1.Purpose:

- **Stored Procedure:**
  - Primarily used for performing an action or a sequence of actions.
  - Can include conditional statements, loops, and transaction control.
  - Can modify data and return output parameters.
- **Function:**
  - Primarily used to encapsulate logic and return a specific value or a table.
  - Cannot include transaction control statements.
  - Cannot modify data directly; designed for read-only operations.

2.Return Values:

- **Stored Procedure:**
  - Can return multiple result sets.
  - Can have output parameters to return values to the calling environment.
  - Does not have a mandatory return statement.
- **Function:**
  - Returns a single value or a table.
  - Scalar functions return a single value.
  - Table-valued functions return a table.

3.Usage in Queries:

- **Stored Procedure:**
  - Can be invoked using the EXECUTE statement or abbreviated as EXEC.
  - Can be called from other stored procedures or functions.
  - Can be used to perform complex operations and calculations.
- **Function:**
  - Can be used in SQL queries, expressions, and as part of other functions.
  - Can be called from within SELECT, WHERE, and other clauses.
  - Can be used in computed columns, views, and stored procedures.

4.Transaction Management:

- **Stored Procedure:**
  - Can include explicit transaction control statements (BEGIN TRANSACTION, COMMIT, ROLLBACK).
- **Function:**
  - Cannot include explicit transaction control statements.
  - Designed for read-only operations; transaction management is the responsibility of the calling environment.

5.Modification of Data:

- **Stored Procedure:**
  - Can modify data directly using INSERT, UPDATE, DELETE statements.
- **Function:**
  - Cannot modify data directly; read-only operations are encouraged.
  - Can modify data indirectly by being part of an UPDATE statement in a stored procedure.

6.Usage in Views:

- **Stored Procedure:**
  - Cannot be used directly in a view.
- **Function:**
  - Can be used in a view, especially if it's a deterministic scalar function.

Summary

stored procedures are more versatile for performing actions, including modifying data and transaction control, while functions are designed for encapsulating logic and returning specific values or tables. The choice between them depends on the specific requirements of your application.

Comparison Between stored procedure and view

1.Purpose:

- **Stored Procedure:**
  - Primarily used for performing an action or a sequence of actions.
  - Can include conditional statements, loops, and transaction control.
  - Can modify data and return output parameters.
  - Useful for complex data manipulations and business logic.
- **View:**
  - Primarily used to simplify complex queries and provide a virtual table based on the result of a SELECT query.
  - Does not perform actions or modify data directly.
  - Often used for security by limiting access to specific columns or rows.
  - Useful for creating reusable views of data.

2.Return Values:

- **Stored Procedure:**
  - Can return multiple result sets.
  - Can have output parameters to return values to the calling environment.
- **View:**
  - Returns a virtual table based on the result of a SELECT query.
  - Does not have output parameters; it's read-only.

3.Transaction Management:

- **Stored Procedure:**
  - Can include explicit transaction control statements (BEGIN TRANSACTION, COMMIT, ROLLBACK).
  - Suitable for handling complex transaction scenarios.
- **View:**
  - Does not include transaction control statements.
  - Typically used for read-only operations; transaction management is handled by the calling environment.

4.Data Modification:

- **Stored Procedure:**
  - Can modify data directly using INSERT, UPDATE, DELETE statements.
  - Designed for data manipulation and business logic.
- **View:**
  - Does not modify data directly.
  - Can be used in conjunction with triggers to indirectly modify data.

5.Usage in Queries:

- **Stored Procedure:**
  - Can be invoked using the EXECUTE statement or abbreviated as EXEC.
  - Can be called from other stored procedures, functions, or applications.
  - Can be used for complex data manipulations and calculations.
- **View:**
  - Can be used in SELECT, FROM, and JOIN clauses of a query.
  - Simplifies complex queries by encapsulating logic.
  - Often used to create a logical abstraction of the data.

6.Security:

- **Stored Procedure:**
  - Requires explicit permission to execute.
  - Fine-grained control over who can execute and modify the procedure.
- **View:**
  - Requires permission to select from the underlying tables or views.
  - Can limit access to specific columns or rows.

Summary

stored procedures are used for performing actions and business logic, including modifying data and handling transactions. Views, on the other hand, are used to simplify queries, provide a logical abstraction of the data, and enforce security by limiting access to specific columns or rows. The choice between them depends on the specific requirements of your application.