**VIEWS IN SQL SERVER:**

• A view in SQL is just a saved SQL query.

• A view can also be considered as a virtual table.

• Views can be used as a mechanism to implement row and column level security.

* SP\_HELPTEXT.
* ALTER the view.
* DROP the view.

• INSERT WITH VIEWS.

• UPDATE WITH VIEWS.

• DELETE WITH VIEWS.

• We can create a view from single or multiple tables.

**STORED PROCEDURE:**

• A Stored Procedure is a set of SQL statements with an assigned name, which are stored in a RDBMS as a group, so it can be reused and shared by multiple programs.

* **Types of Stored Procedure:**

**•** SystemStored Procedures.

• User Defined Stored Procedures.

**COMMON TABLE EXPRESSION (CTE):**

• CTE is introduced in SQL Server 2005.

• A CTE allows us to define a temporary result set, that can be linked immediately with the **SELECT, INSERT, UPDATE or DELETE** statement.

• CTE is like a temporary result set defined within the execution scope of a single SELECT, INSERT, UPDATE, DELETE or CREATE VIEW statement.

• The CTE can also be used in a view.

• A CTE is defined at the start of query and can be referenced several times in outer query.

• Key advantages of CTEs are improved readability and ease in maintenance of complex queries.

• CTE exists in memory only while the query is running. After the query is run, the query is discarded; it can’t be used for the next SQL query unless we define it again. Still, the same CTE might be referenced several times in the main query and any subqueries.

• We can create multiple CTEs by using single WITH clause.

• CTEs are limited in scope to the execution of the outer query. Hence, when the outer query ends, the lifetime of CTE will end.

• We need to define a name for CTE and, define unique names for each of the columns referenced in the SELECT clause of the CTE.

• It is possible to use inline or external aliases for columns in CTEs.

• A single CTE can be referenced multiple times in the same query with one definition.

• Multiple CTEs can also be defined in the same with clause.

**Difference between View and CTE:**

• A view is stored SQL query that is executed each time you reference it in another query. Note that a view doesn’t store the output of a particular query- It stores the query itself.

• The key thing to remember about SQL views is that, in contrast to a CTE, a view is physical object in a database and is stored on a disk.

•However, views store the query only, not the data returned by the query. The data is computed each time you reference the view in your query.

**CTE VS SUBQUERY:**

• CTE is created before the outer query.

• Sub query is created after the outer query.