**CURSOR IN SQL SERVER:**

* Cursor is a temporary memory or temporary workstation.
* A SQL cursor is a database object that is used to retrieve data from a result set one row at a time.
* A SQL cursor is used when the data needs to be updated row by row.
* It is allocated by database server at the time of performing DML operations on table by user.
* Cursors are used to store database tables.
* It allows you to process individual row returned by a query.

**Types of Cursors:**

1. **Implicit Cursor:** Implicit Cursors are also known as **Default Cursor** of SQL Server. These cursors are allocated by SQL Server when the user performs DML operations.
2. **Explicit Cursor:** Explicit Cursors are created by users whenever the user requires them. Explicit cursors are used for fetching data from table in row-by-row manner.

**Methods of Cursor:**

* NEXT.
* PRIOR.
* FIRST.
* LAST.
* ABSOLUTE N.
* RELATIVE N.

**Cursor Lifecycle:**

1. **Declare Cursor:** A cursor is declared by defining the SQL statement.

* We can declare a cursor by specifying its name with the data type CURSOR after the **DECLARE** keyword. Then, we will write the SELECT statement that defines the output for the cursor.
* **Syntax: DECLARE cursor\_name CURSOR FOR select\_statement;**

1. **Opening Cursor:** A cursor is opened for storing data retrieved from the result set.

* It's a second step in which we open the cursor to store data retrieved from the result set. We can do this by using the below SQL statement:
* **Syntax: OPEN cursor\_name;**

1. **Fetching Cursor:** When a cursor is opened, rows can be fetched from the cursor one by one or in a block to do data manipulation.

* It's a third step in which rows can be fetched one by one or in a block to do data manipulation like insert, update, and delete operations on the currently active row in the cursor. We can do this by using the below SQL statement:
* **FETCH** **NEXT** **FROM** **cursor** **INTO** **variable\_list;**
* We can also use the **@@FETCHSTATUS function** in SQL Server to get the status of the most recent FETCH statement cursor that was executed against the cursor. The **FETCH** statement was successful when the @@FETCHSTATUS gives zero output. The **WHILE** statement can be used to retrieve all records from the cursor. The following code explains it more clearly:
* **WHILE @@FETCH\_STATUS = 0**

**BEGIN**

**FETCH** **NEXT** **FROM cursor\_name;**

**END**;

1. **Closing Cursor:** The cursor should be closed explicitly after data manipulation.

* It's a fourth step in which the cursor should be closed after we finished work with a cursor. We can do this by using the below SQL statement:
* **Syntax: CLOSE cursor\_name;**

1. **De-allocating Cursor:** Cursors should be deallocated to delete cursor definition and release all the system resources associated with the cursor.

* It is the fifth and final step in which we will erase the cursor definition and release all the system resources associated with the cursor. We can do this by using the below SQL statement:
* **Syntax: DEALLOCATE cursor\_name;**
* **We can use cursors in 2 ways:**
* With Cursor Variables.
* Without Cursor Variables.