Life Cycle of Cursor

1. Declare Cursor

A cursor is declared by defining the SQL statement that returns a result set.

1. Open

A Cursor is opened and populated by executing the SQL statement defined by the cursor.

1. Fetch

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

1. Close

After data manipulation, we should close the cursor explicitly.

1. DE allocate

Finally, we need to delete the cursor definition and released all the system resources associated with the cursor.

### Syntax to Declare Cursor

The basic syntax is given below

1. **DECLARE cursor\_name CURSOR**
2. **[LOCAL | GLOBAL] *--define cursor scope***
3. **[FORWARD\_ONLY | SCROLL] *--define cursor movements (forward/backward)***
4. **FOR select\_statement *--define SQL Select statement***

### Syntax to Open Cursor

A Cursor can be opened locally or globally. By default it is opened locally. The basic syntax to open cursor is given below:

1. **OPEN [GLOBAL] cursor\_name *--by default it is local***

### Syntax to Fetch Cursor

Fetch statement provides the many options to retrieve the rows from the cursor. NEXT is the default option. The basic syntax to fetch cursor is given below:

1. **FETCH [NEXT|PRIOR|FIRST|LAST|ABSOLUTE n|RELATIVE n]**
2. **FROM cursor name**
3. **INTO @Variable\_name [1,2,..n]**

### Syntax to Close Cursor

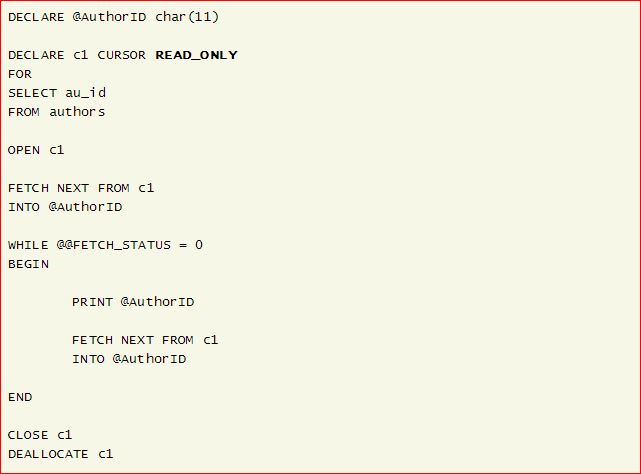
Close statement closed the cursor explicitly. The basic syntax to close cursor is given below:

1. **CLOSE cursor\_name *--after closing it can be reopen***

### Syntax to Deallocate Cursor

Deallocate statement delete the cursor definition and free all the system resources associated with the cursor. The basic syntax to close cursor is given below:

1. **DEALLOCATE cursor\_name *--after deallocation it can't be reopen***



DECLARE @fName varchar(50), @lName varchar(50)

DECLARE cursorName CURSOR *-- Declare cursor*

LOCAL SCROLL STATIC

FOR

Select firstName, lastName FROM myTable

OPEN cursorName *-- open the cursor*

FETCH NEXT FROM cursorName

INTO @fName, @lName

PRINT @fName + ' ' + @lName *-- print the name*

WHILE @@FETCH\_STATUS = 0

BEGIN

FETCH NEXT FROM cursorName

INTO @fName, @lName

PRINT @fName + ' ' + @lName *-- print the name*

END

CLOSE cursorName *-- close the cursor*

DEALLOCATE cursorName *-- Deallocate the cursor*