SQL Server CURSOR

Summary: in this tutorial, you will learn how to use the SQL Server cursor to process a result set, one row at a time.

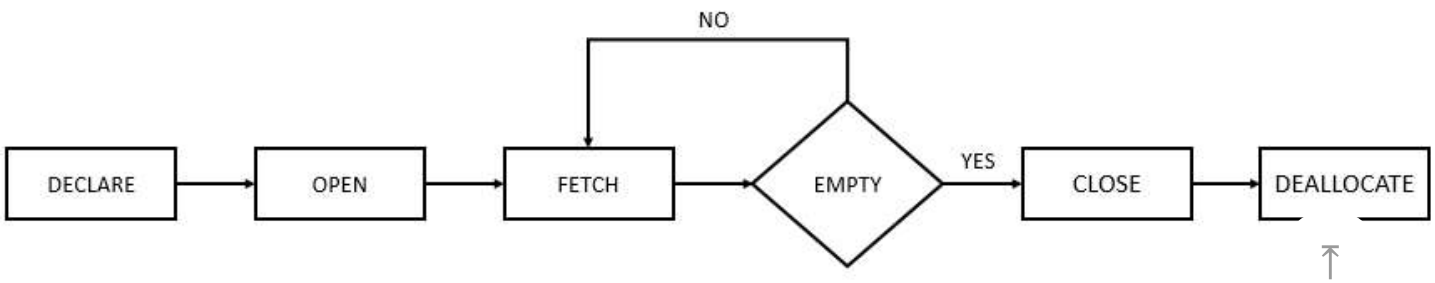
SQL works based on set e.g., `SELECT` statement returns a set of rows which is called a result set. However, sometimes, you may want to process a data set on a row by row basis. This is where cursors come into play.

What is a database cursor

A database cursor is an object that enables traversal over the rows of a result set. It allows you to process individual row returned by a query.

SQL Server cursor life cycle

These are steps for using a cursor:



First, declare a cursor.

```
DECLARE cursor_name CURSOR
    FOR select_statement;
```

To declare a cursor, you specify its name after the `DECLARE` keyword with the `CURSOR` data type and provide a `SELECT` statement that defines the result set for the cursor.

Next, open and popular the cursor by executing the `SELECT` statement:

```
OPEN cursor_name;
```

Then, fetch a row from the cursor into one or more variables:

```
FETCH NEXT FROM cursor INTO variable_list;
```

SQL Server provides the `@@FETCHSTATUS` function that returns the status of the last cursor `FETCH` statement executed against the cursor; If `@@FETCHSTATUS` returns 0, meaning the `FETCH` statement was successful. You can use the `WHILE` statement to fetch all rows from the cursor as shown in the following code:

```
WHILE @@FETCH_STATUS = 0
    BEGIN
        FETCH NEXT FROM cursor_name;
    END;
```

After that, close the cursor:

```
CLOSE cursor_name;
```

Finally, deallocate the cursor:

```
DEALLOCATE cursor_name;
```

SQL Server cursor example

First, declare two variables to hold product name and list price, and a cursor to hold the result of a query that selects product name and list price from the `production.products` table:

```
DECLARE
    @product_name VARCHAR(MAX),
    @list_price   DECIMAL;

```

```
DECLARE cursor_product CURSOR
FOR SELECT
        product_name,
        list_price
    FROM
        production.products;
```

Next, open the cursor:

```
OPEN cursor_product;
```

Then, fetch each row from the cursor and print out the product name and list price:

```
FETCH NEXT FROM cursor_product INTO
    @product_name,
    @list_price;

WHILE @@FETCH_STATUS = 0
    BEGIN
        PRINT @product_name + CAST(@list_price AS varchar);
        FETCH NEXT FROM cursor_product INTO
            @product_name,
            @list_price;
    END;
```

After that, close the cursor.

```
CLOSE cursor_product;
```

Finally, deallocate the cursor to release it.

```
DEALLOCATE cursor_product;
```

The following code snippets put everything together:

```
DECLARE
    @product_name VARCHAR(MAX),
    @list_price   DECIMAL;

DECLARE cursor_product CURSOR
FOR SELECT
        product_name,
        list_price
    FROM
        production.products;

OPEN cursor_product;

```

```
FETCH NEXT FROM cursor_product INTO
    @product_name,
    @list_price;

WHILE @@FETCH_STATUS = 0
    BEGIN
        PRINT @product_name + CAST(@list_price AS varchar);
        FETCH NEXT FROM cursor_product INTO
            @product_name,
            @list_price;
    END;

CLOSE cursor_product;

DEALLOCATE cursor_product;
```

Here is the partial output:

```
Trek 820 - 2016380
Ritchey Timberwolf Frameset - 2016750
Surly Wednesday Frameset - 20161000
Trek Fuel EX 8 29 - 20162900
Heller Shagamaw Frame - 20161321
Surly Ice Cream Truck Frameset - 2016470
Trek Slash 8 27.5 - 20164000
Trek Remedy 29 Carbon Frameset - 20161800
Trek Conduit+ - 20163000
Surly Straggler - 20161549
Surly Straggler 650b - 20161681
Electra Townie Original 21D - 2016550
```

In this tutorial, you have learned how to use the SQL Server cursor to process a result set, each row at a time.

« Previous Tutorial: SQL Server CONTINUE

Next Tutorial: SQL Server TRY CATCH »

Search this website

GETTING STARTED

What is SQL Server

Install the SQL Server

Connect to the SQL Server

SQL Server Sample Database

Load Sample Database

DATA MANIPULATION

SELECT

ORDER BY

OFFSET FETCH

SELECT TOP

SELECT DISTINCT

WHERE

NULL

AND

OR

IN

DELETE

MERGE

PIVOT

DATA DEFINITION

Create New Database

Drop Database

Create Schema

Alter Schema

Drop Schema

Create New Table

Identity Column

Sequence

Add Column

Modify Column

Drop Column

Computed Columns

Rename Table

Drop Table

Truncate Table

Temporary Tables

Synonym

SELECT INTO

PRIMARY KEY

FOREIGN KEY

CHECK Constraint

UNIQUE Constraint

NOT NULL Constraint

DATA TYPES

Data Types

BIT

CHAR

DATE

DATETIME2

DATETIMEOFFSET

Decimal

INT

NCHAR

NVARCHAR

TIME

VARCHAR

EXPRESSIONS

CASE

COALESCE

NULLIF

ABOUT SQLSERVERTUTORIAL.NET

SQLServerTutorial.net website designed for Developers, Database Administrators, and Solution Architects who want to get started SQL Server quickly.

Search this website

RECENT TUTORIALS

SQL Server Synonym

SQL Server DROP SCHEMA

SQL Server ALTER SCHEMA

SQL Server CREATE SCHEMA

SQL Server SYSDATETIMEOFFSET Function

SITE LINKS

About

Contact

Privacy Policy

Terms of Use