T-SQL: A Simple Example Using a Cursor

For more information on cursors, also take a look at the free SQL query training provided by Steve Stedman.

In SQL Server the cursor is a tool that is used to iterate over a result set, or to loop through each row of a result set one row at a time. It may not be the best way to work with a set of data, but if you need to loop row by agonizing row (RBAR) in a T-SQL script then a cursor is one way of doing it.

Note: If you are new to SQL Server and come from an Oracle background, you should know that cursors on SQL Server are different from those on Oracle.

Before creating the cursor, we will just start with a simple query that will end up being used in the cursor.

Which looks something like this:

SQLQuery7.sql - vm2008r2.AdventureWorks2008 (sa (57))*

```
use AdventureWorks2008;
GO

SELECT BusinessEntityID, Name
  FROM Sales.Store;
```

100 %

Results | Messages

	BusinessEntityID	Name
1	292	Next-Door Bike Store
2	294	Professional Sales and Service
3	296	Riders Company
4	298	The Bike Mechanics
5	300	Nationwide Supply
6	302	Area Bike Accessories
7	304	Bicycle Accessories and Kits
8	306	Clamps & Brackets Co.
9	308	Valley Bicycle Specialists
10	310	New Bikes Company
11	312	Vinyl and Plastic Goods Corporation
12	314	Top of the Line Bikes
13	316	Fun Toys and Bikes
14	318	Great Bikes
15	320	Metropolitan Sales and Rental
16	322	Irregulars Outlet
17	324	Valley Toy Store
18	326	Worthwhile Activity Store
19	328	Purchase Mart
20	330	Major Sport Suppliers

Query exe... | vm2008r2 (10.50 RTM) | sa (57) | AdventureWorks2008 | 00:00:00 | 701 rows

Now to convert it to a cursor, instead of just a select statement.

Step 1: Declare variables to hold the output from the cursor.

Step 2: Declare the cursor object;

Step 3: Assign the query to the cursor.

Step 4: Open the cursor.

Step 5: Fetch the first row.

Step 5: Loop until there are no more results. In the loop print out the ID and the name from the result set and fetch the net row.

Step 6: Close the cursor.

Step 7: Deallocate the cursor to free up any memory or open result sets.

Now putting it all together:

SQLQuery7.sql - vm2008r2.AdventureWorks2008 (sa (57))*

```sql
DECLARE @BusinessEntityID as INT;
DECLARE @BusinessName as NVARCHAR(50);

DECLARE @BusinessCursor as CURSOR;

SET @BusinessCursor = CURSOR FOR
SELECT BusinessEntityID, Name
  FROM Sales.Store;

OPEN @BusinessCursor;
FETCH NEXT FROM @BusinessCursor INTO @BusinessEntityID, @BusinessName;

WHILE @@FETCH_STATUS = 0
BEGIN
    PRINT cast(@BusinessEntityID as VARCHAR (50)) + ' ' + @BusinessName;
    FETCH NEXT FROM @BusinessCursor INTO @BusinessEntityID, @BusinessName;
END

CLOSE @BusinessCursor;
DEALLOCATE @BusinessCursor;
```

100 %

Messages

```
292 Next-Door Bike Store
294 Professional Sales and Service
296 Riders Company
298 The Bike Mechanics
300 Nationwide Supply
302 Area Bike Accessories
304 Bicycle Accessories and Kits
306 Clamps & Brackets Co.
308 Valley Bicycle Specialists
310 New Bikes Company
312 Vinyl and Plastic Goods Corporation
314 Top of the Line Bikes
316 Fun Toys and Bikes
318 Great Bikes
320 Metropolitan Sales and Rental
```

100 %

Query executed successfully. | vm2008r2 (10.50 RTM) | sa (57) | AdventureWorks2008 | 00:00:00 | 0 rows

This should give you a quick overview of how to quickly build and use a cursor on Microsoft SQL Server. The example shown was run on SQL Server 2008, and works the same on SQL Server 2005 , SQL Server 2008R2, SQL Server 2012 or SQL Server 2014.

Here is a video showing a similar overview of using cursors in TSQL.

Enjoy!

-Steve Stedman

Related Links

- Video: Simple Introduction to TSQL Cursors
- Free SQL Query Training for the 70-461 course

- Simple Cursor Example : FORWARD_ONLY vs FAST FORWARD
- Video Training on Using Cursors With SQL Server
- More details on Cursors
- Using a CURSOR to list databases
- Cursors - Technical debt
- Listing Database Example with a CURSOR