

Modifying Data Using T-SQL Statements



Xavier Morera

HELPING DEVELOPERS UNDERSTAND SEARCH & BIG DATA

@xmorera www.xaviermorera.com



Modifying Data Using T-SQL Statements

INSERT

UPDATE

DELETE

Add Data

Modify Data

Remove Data



SQL Server 2017 ▾


Filter by title

- > Spatial geometry & instances (geometry Data Type)
- > Data types
- > XML
- > DBCC
- > Functions
- > Language elements
- > Queries
- ▾ Statements
 - Statements
 - ▾ General
 - BULK INSERT
 - DELETE
 - DISABLE TRIGGER
 - ENABLE TRIGGER
 - INSERT
 - INSERT (SQL Graph)
 - UPDATE**
 - MERGE
 - TRUNCATE TABLE
 - UPDATE STATISTICS
 - > ALTER
 - > Backup & restore
 - > CREATE
 - > Collations
 - > DROP

Download PDF

UPDATE (Transact-SQL)

09/05/2017 • 38 minutes to read • 🧑🏿 🧑🏻 🧑🏾 🧑🏽 +7

APPLIES TO: ✓ SQL Server ✓ Azure SQL Database ✓ Azure SQL Data Warehouse ✓ Parallel Data WarehouseChanges existing data in a table or view in SQL Server 2017. For examples, see [Examples](#). [Transact-SQL Syntax Conventions](#)

Syntax

SQL

Copy

```
-- Syntax for SQL Server and Azure SQL Database

[ WITH <common_table_expression> [...n] ]
UPDATE
    [ TOP ( expression ) [ PERCENT ] ]
    { { table_alias | <object> | rowset_function_limited
      [ WITH ( <Table_Hint_Limited> [ ...n ] ) ]
    }
    | @table_variable
  }
SET
    { column_name = { expression | DEFAULT | NULL }
    | { udt_column_name. { { property_name = expression
                          | field_name = expression }
                        | method_name ( argument [ ,...n ] )
                      }
    }
    | column_name { .WRITE ( expression , @Offset , @Length ) }
    | @variable = expression
    | @variable = column = expression
    | column_name { += | -= | *= | /= | % = | &= | ^= | |= } expression
    | @variable { += | -= | *= | /= | % = | &= | ^= | |= } expression
    | @variable = column { += | -= | *= | /= | % = | &= | ^= | |= } expression
  } [ ,...n ]

[ <OUTPUT Clause> ]
[ FROM( <table_source> ) [ ,...n ] ]
[ WHERE { <search_condition>
        | { [ CURRENT OF
            { { [ GLOBAL ] cursor_name
              | { [ LOCAL ] cursor_name
            }
          }
        }
      ]
    ]
```

Thank you.

In this article

[Syntax](#)[Arguments](#)[Best Practices](#)[Compatibility Support](#)[Data Types](#)[Error Handling](#)[Interoperability](#)[Limitations and Restrictions](#)[Locking Behavior](#)[Logging Behavior](#)[Security](#)[Examples](#)[Examples: SQL Data Warehouse and Parallel Data Warehouse](#)[See Also](#)

UPDATE Syntax

Specify destination
table

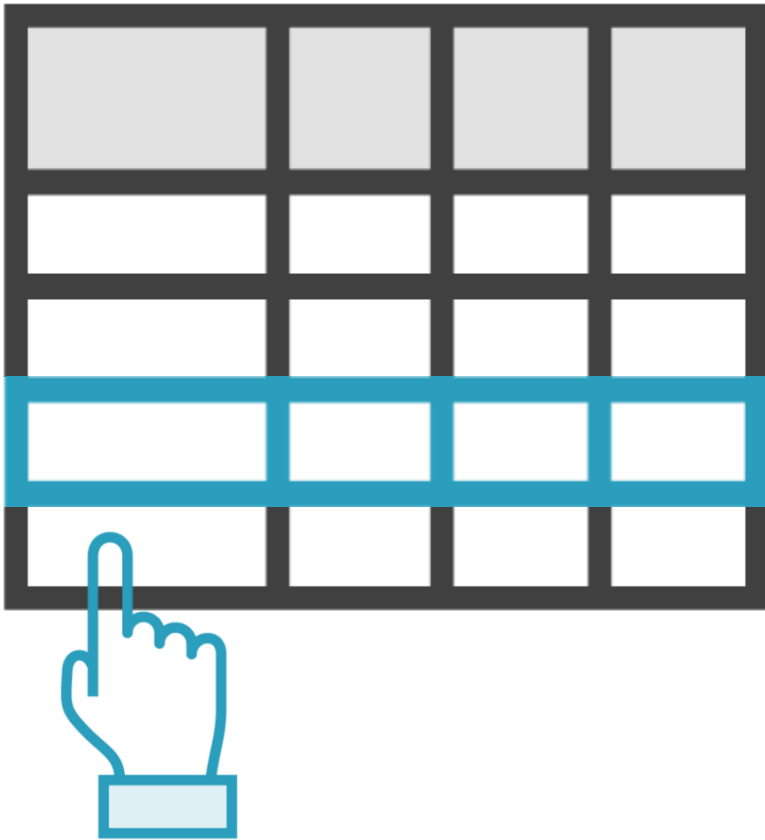
Which columns are going to be
updated

```
UPDATE table_name SET (column_1 = value_1, column_2 = value_2, ..., column_n = value_n)  
WHERE { condition }
```

Specify search condition for
which records should be
modified



Using UPDATE to Modify Data in One Row



One of the most basic scenarios

- Modifying data in one row
- One or many columns

Narrow down scope using WHERE

Careful with

- Autoincremented values
- Constraints

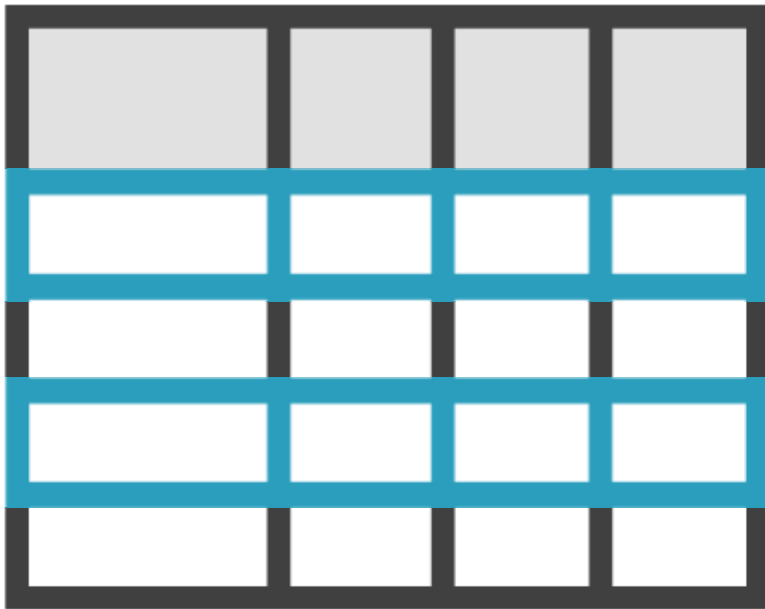
Demo



Using UPDATE to Modify Data
in One Row



Using UPDATE to Modify Data in Multiple Rows



Modify data in many rows

- All at once
- Set a new value or a calculated value

Limit number of records with WHERE

- Or even modify data in the entire table

Demo



Using UPDATE to Modify Data
in Multiple Rows



Retrieving Number of Affected Rows

2

SSSMS tells us how many rows are affected

But an application may need a
programmatic way

For obtaining rows affected

Use @@ROWCOUNT



Demo



Using @@ROWCOUNT to Obtain Number of Rows Affected by UPDATE



Modifying Data Using UPDATE with Constraints



Updating values in a column is straightforward

Unless there is a constraint

Like a foreign key

An error is raised if a constraint is violated

Demo



Using UPDATE with Constraints



UPDATE Using Variables

$\{y, x\}$

Assigning values to columns

You can also assign values to variables

- Perform operations
- Use them for UPDATE

Variable assignment performed on each row in the output

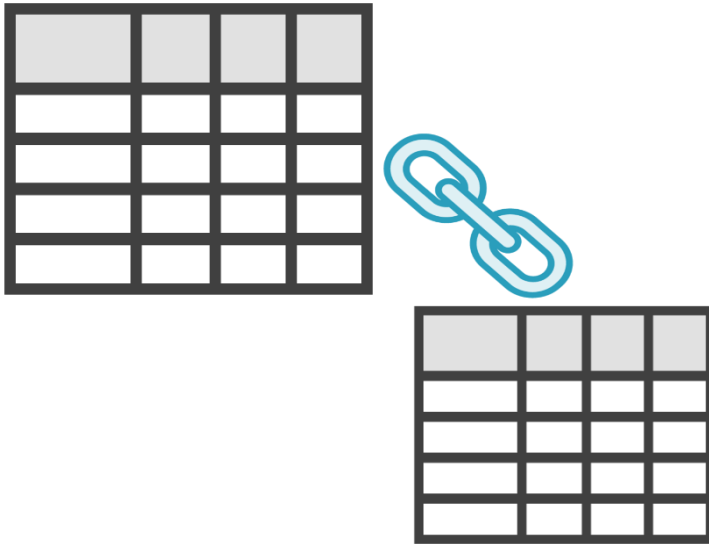
Demo



Using UPDATE with Variables



Using FROM within an UPDATE Statement



Values and variable

Update a column based on data in a different table

- FROM
- JOIN

Demo



Using FROM and JOIN within an
UPDATE Statement



SQL Server 2017 ▾

Filter by title

Statements

General

BULK INSERT

DELETE

DISABLE TRIGGER

ENABLE TRIGGER

INSERT

INSERT (SQL Graph)

UPDATE

MERGE

TRUNCATE TABLE

UPDATE STATISTICS

> ALTER

> Backup & restore

> CREATE

> Collations

> DROP

> Permissions

> Service Broker





> SET

> xQuery


Download PDF

UPDATE (Transact-SQL)

09/05/2017 • 38 minutes to read •  +7

APPLIES TO:  SQL Server  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

Changes existing data in a table or view in SQL Server 2017. For examples, see [Examples](#).

 [Transact-SQL Syntax Conventions](#)

Syntax

SQL

Copy

-- Syntax for SQL Server and Azure SQL Database

```
[ WITH <common_table_expression> [...n] ]
UPDATE
    [ TOP ( expression ) [ PERCENT ] ]
    { { table_alias | <object> | rowset_function_limited
      [ WITH ( <Table_Hint_Limited> [ ...n ] ) ]
    }
    | @table_variable
  }
SET
    { column_name = { expression | DEFAULT | NULL }
      | { udt_column_name. { { property_name = expression
                           | field_name = expression }
                           | method_name ( argument [ ,...n ] )
                        }
    }
    | column_name { .WRITE ( expression , @Offset , @Length ) }
    | @variable = expression
    | @variable = column = expression
    | column_name { += | -= | *= | /= | %= | &= | ^= | |= } expression
    | @variable { += | -= | *= | /= | %= | &= | ^= | |= } expression
```

Thank you.

In this article

[Syntax](#)

[Arguments](#)

[Best Practices](#)

[Compatibility Support](#)

[Data Types](#)

[Error Handling](#)

[Interoperability](#)

[Limitations and Restrictions](#)

[Locking Behavior](#)

[Logging Behavior](#)

[Security](#)

[Examples](#)

[Examples: SQL Data Warehouse and Parallel Data Warehouse](#)

[See Also](#)

Takeaway



Modify data using UPDATE

UPDATE data in

- One row
- Multiple rows

@@ROWCOUNT and **OUTPUT**

Use variables to perform operations

FROM to use other tables and views