

## OUTPUT Clause (Transact-SQL)

Returns information from, or expressions based on, each row affected by an INSERT, UPDATE, DELETE, or MERGE statement. These results can be returned to the processing application for use in such things as confirmation messages, archiving, and other such application requirements. The results can also be inserted into a table or table variable. Additionally, one can capture the results of an OUTPUT clause in a nested INSERT, UPDATE, DELETE, or MERGE statement, and insert those results into a target table or view.

Used in: DELETE; INSERT; UPDATE; MERGE

The OUTPUT clause – helps to get the identity values from a large amount of data that have been inserted in to the database. The OUTPUT clause is a part of the query that will return data from before or after the operations is completed. For example, one has inserted data in to a table and want the ID column values (which are auto-numbers). The OUTPUT clause gives this information!

To access the data that is being altered in the sql statement, one need to use special column prefixes that SQL Server provide. The two special prefixes are “**inserted**” and “**deleted**”. During an insert statement, the **inserted** prefix is available to use. During a delete statement, the **deleted** prefix is available to use. During the update and merge statements, both the **deleted** and **inserted** prefixes are available to use. In these cases, the deleted represents the data before it was changed and the inserted represents the data after it was changed.

*One thing to note is that the data being outputted must go into a table or table variable.*

We consider the Authors table

id	name
4	J.K. Rowling
5	transaction 2
6	Petre Ispirescu t...
7	test
NULL	NULL

### OUTPUT Clause On An INSERT Statement

```
-- OUTPUT Clause On An INSERT Statement
DECLARE @OutputData TABLE
        (AuthorsID INT,
         AuthorsName VARCHAR(50))

-- Insert into the table and stick the resulting
Authors name and identity column in to a temp variable
INSERT
INTO Authors (name)
OUTPUT inserted.id, inserted.name
INTO @OutputData (AuthorsID, AuthorsName)
VALUES ('Mircea Eliade'), ('Dan Brown')

-- View the inserted data
SELECT *
```

	AuthorsID	AuthorsName
1	8	Mircea Eliade
2	9	Dan Brown

```
FROM @OutputData

select * from Authors
```

Results		
	id	name
1	4	J.K. Rowling
2	5	transaction 2
3	6	Petre Ispirescu transaction 1
4	7	test
5	8	Mircea Eliade
6	9	Dan Brown

## OUTPUT Clause On A DELETE Statement

```
-- OUTPUT Clause On A DELETE Statement
DECLARE @OutputData TABLE
        (AuthorsID INT,
         AuthorsName VARCHAR(50))

DELETE
FROM Authors
OUTPUT deleted.id, deleted.name
INTO @OutputData (AuthorsID, AuthorsName)
WHERE name = 'Mircea Eliade'

-- View the deleted data
SELECT *
FROM @OutputData

select * from Authors
```

Results		
	AuthorsID	AuthorsName
1	8	8

  

	id	name
1	4	J.K. Rowling
2	5	transaction 2
3	6	Petre Ispirescu transaction 1
4	7	test
5	9	Dan Brown

Query executed successfully.

## OUTPUT Clause On An UPDATE Statement

```
-- OUTPUT Clause On An UPDATE Statement
DECLARE @OutputData TABLE
        (AuthorsID INT,
         OldAuthorsName
         VARCHAR(50) ,
         NewAuthorsName
         VARCHAR(50))

-- Update the table and output the Authors
name and identity column to the temp table
UPDATE Authors
SET name = 'NEW NAME'
OUTPUT inserted.id, deleted.name,
inserted.name
INTO @OutputData
        (AuthorsID, OldAuthorsName,
         NewAuthorsName)
WHERE name = 'Dan Brown'

select * from Authors
```

Results		
	id	name
1	4	J.K. Rowling
2	5	transaction 2
3	6	Petre Ispirescu transaction 1
4	7	test
5	9	NEW NAME

## OUTPUT Clause On A MERGE Statement

```

-- OUTPUT Clause On A MERGE Statement
DECLARE      @OutputData  TABLE
              (AuthorsID  INT,
               OldAuthorsName  VARCHAR(50) ,
               NewAuthorsName  VARCHAR(50))

DECLARE      @ChangeData  TABLE
              (OldAuthorsName  VARCHAR(50),
               NewAuthorsName  VARCHAR(50))

INSERT INTO  @ChangeData VALUES ('NEW NAME', 'Dan Brown MERGE')
-- Update the table and output the Authors name and identity
column to the temp table
MERGE Authors AS target
USING @ChangeData AS source
      ON (source.OldAuthorsName = target.name)
WHEN MATCHED
THEN UPDATE
      SET target.name = source.NewAuthorsName
OUTPUT inserted.id, deleted.name, inserted.name
INTO @OutputData (AuthorsID, OldAuthorsName, NewAuthorsName);

-- View the updated data
SELECT *
FROM @OutputData

select * from Authors

```

Results		Messages	
	AuthorsID	OldAuthorsName	NewAuthorsName
1	9	NEW NAME	Dan Brown MERGE

  

	id	name
1	4	J.K. Rowling
2	5	transaction 2
3	6	Petre Ispirescu transaction 1
4	7	test
5	9	Dan Brown MERGE

References:

<http://www.tech-recipes.com/rx/47032/a-beginners-guide-to-the-output-clause-in-sql-server/>

<http://www.made2mentor.com/2013/06/using-the-output-clause-with-t-sql-merge/>

<http://www.sqlservercentral.com/articles/T-SQL/156901/>

<https://stackoverflow.com/questions/23240274/how-to-retrieve-old-values-in-output-clause-with-sql-merge-statement>