

IDENTITY (Function) (Transact-SQL)

Article • 12/30/2022

Applies to:  SQL Server  Azure SQL Managed Instance

Is used only in a SELECT statement with an INTO *table* clause to insert an identity column into a new table. Although similar, the IDENTITY function is not the IDENTITY property that is used with CREATE TABLE and ALTER TABLE.

Note

To create an automatically incrementing number that can be used in multiple tables or that can be called from applications without referencing any table, see [Sequence Numbers](#).

 [Transact-SQL syntax conventions](#)

Syntax

syntaxsql

```
IDENTITY (data_type [ , seed , increment ] ) AS column_name
```

Note

To view Transact-SQL syntax for SQL Server 2014 (12.x) and earlier versions, see [Previous versions documentation](#).

Arguments

data_type

Is the data type of the identity column. Valid data types for an identity column are any data types of the integer data type category, except for the **bit** data type, or **decimal** data type.

seed

Is the integer value to be assigned to the first row in the table. Each subsequent row is

assigned the next identity value, which is equal to the last IDENTITY value plus the *increment* value. If neither *seed* nor *increment* is specified, both default to 1.

increment

Is the integer value to add to the *seed* value for successive rows in the table.

column_name

Is the name of the column that is to be inserted into the new table.

Return Types

Returns the same as *data_type*.

Remarks

Because this function creates a column in a table, a name for the column must be specified in the select list in one of the following ways:

SQL

```
--(1)
SELECT IDENTITY(int, 1,1) AS ID_Num
INTO NewTable
FROM OldTable;

--(2)
SELECT ID_Num = IDENTITY(int, 1, 1)
INTO NewTable
FROM OldTable;
```

Examples

The following example inserts all rows from the `Contact` table from the AdventureWorks2022database into a new table called `NewContact`. The IDENTITY function is used to start identification numbers at 100 instead of 1 in the `NewContact` table.

SQL

```
USE AdventureWorks2022;
GO
IF OBJECT_ID (N'Person.NewContact', N'U') IS NOT NULL
    DROP TABLE Person.NewContact;
GO
```

```
ALTER DATABASE AdventureWorks2022 SET RECOVERY BULK_LOGGED;  
GO  
SELECT IDENTITY(smallint, 100, 1) AS ContactNum,  
       FirstName AS First,  
       LastName AS Last  
INTO Person.NewContact  
FROM Person.Person;  
GO  
ALTER DATABASE AdventureWorks2022 SET RECOVERY FULL;  
GO  
SELECT ContactNum, First, Last FROM Person.NewContact;  
GO
```

See Also

[CREATE TABLE \(Transact-SQL\)](#)

[@@IDENTITY \(Transact-SQL\)](#)

[IDENTITY \(Property\) \(Transact-SQL\)](#)

[SELECT @local_variable \(Transact-SQL\)](#)

[DBCC CHECKIDENT \(Transact-SQL\)](#)

[sys.identity_columns \(Transact-SQL\)](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)