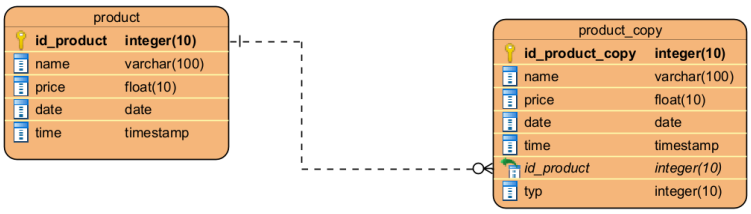
Trigger with tables inserted and deleted

Example

Write a trigger that records insert and delete changes to the product table.



Variables are declared in the body of a batch or procedure with the DECLARE statement and are assigned values by using either a SET or SELECT statement. Cursor variables can be declared with this statement and used with other cursor-related statements. After declaration, all variables are initialized as NULL, unless a value is provided as part of the declaration.

Table is a special data type used to store a result set for processing at a later time. table is primarily used for temporarily storing a set of rows that are returned as the table-valued function result set. Functions and variables can be declared to be of type table. table variables can be used in functions, stored procedures, and batches. To declare variables of type table, use DECLARE @local_variable.

DML trigger statements use two special tables: the deleted and inserted tables. SQL Server automatically creates and manages these tables. You can use these temporary, memory-resident tables to test the effects of certain data modifications and to set conditions for DML trigger actions. You cannot directly modify the data in the tables or perform data definition language (DDL) operations on the tables, such as CREATE INDEX.

In DML triggers, the inserted and deleted tables are primarily used to perform the following:

Extend referential integrity between tables.

Insert or update data in base tables underlying a view.

Test for errors and take action based on the error.

Find the difference between the state of a table before and after a data modification and take actions based on that difference.

The deleted table stores copies of the affected rows in the trigger table before they were changed by a DELETE or UPDATE statement (the trigger table is the table on which the DML trigger runs). During the execution of a DELETE or UPDATE statement, the affected rows are first copied from the trigger table and transferred to the deleted table.

The inserted table stores copies of the new or changed rows after an INSERT or UPDATE statement. During the execution of an INSERT or UPDATE statement, the new or changed rows in the trigger table are copied to the inserted table. The rows in the inserted table are copies of the new or updated rows in the trigger table.

An update transaction is similar to a delete operation followed by an insert operation. During the execution of an UPDATE statement, the following sequence of events occurs:

The original row is copied from the trigger table to the deleted table.

The trigger table is updated with the new values from the UPDATE statement.

The updated row in the trigger table is copied to the inserted table.

This allows you to compare the contents of the row before the update (in the deleted table) with the new row values after the update (in the inserted table).

When you set trigger conditions, use the inserted and deleted tables appropriately for the action that fired the trigger. Although referencing the deleted table when testing an INSERT or the inserted table when testing a DELETE does not cause any errors, these trigger test tables do not contain any rows in these cases.

Note

If trigger actions depend on the number of rows a data modification effects, use tests (such as an examination of @@ROWCOUNT) for multirow data modifications (an INSERT, DELETE, or UPDATE based on a SELECT statement), and take appropriate actions.