SQL SERVER – @@IDENTITY vs SCOPE\_IDENTITY() vs IDENT\_CURRENT – Retrieve Last Inserted Identity of Record

SELECT @@IDENTITY

It returns the last IDENTITY value produced on a connection, regardless of the table that produced the value, and regardless of the scope of the statement that produced the value.

@@IDENTITY will return the last identity value entered into a table in your current session. While @@IDENTITY is limited to the current session, it is not limited to the current scope. If you have a trigger on a table that causes an identity to be created in another table, you will get the identity that was created last, even if it was the trigger that created it.

SELECT SCOPE\_IDENTITY()

It returns the last IDENTITY value produced on a connection and by a statement in the same scope, regardless of the table that produced the value.

SCOPE\_IDENTITY(), like @@IDENTITY, will return the last identity value created in the current session, but it will also limit it to your current scope as well. In other words, it will return the last identity value that you explicitly created, rather than any identity that was created by a trigger or a user defined function.

SELECT IDENT\_CURRENT(‘tablename’)

It returns the last IDENTITY value produced in a table, regardless of the connection that created the value, and regardless of the scope of the statement that produced the value.

IDENT\_CURRENT is not limited by scope and session; it is limited to a specified table. IDENT\_CURRENT returns the identity value generated for a specific table in any session and any scope.

To avoid the potential problems associated with adding a trigger later on, always use SCOPE\_IDENTITY() to return the identity of the recently added row in your T SQL Statement or Stored Procedure.