

# Domine Microsoft SQL Server 2008



**César Pérez López**

**Alfaomega**  **Ra-Ma®**

## 11.3 ELIMINAR DATOS. LAS SENTENCIAS DELETE Y TRUNCATE

En el caso de la eliminación de filas de datos en tablas ya existentes, la sentencia básica es DELETE.

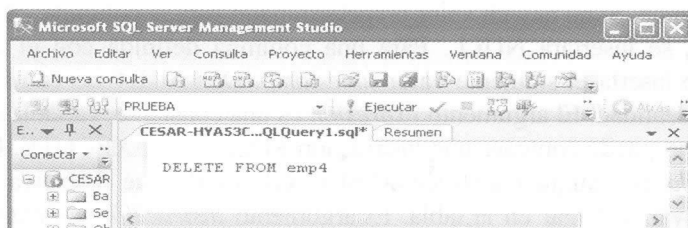
### 11.3.1 Eliminar filas con DELETE

La sentencia DELETE quita una o varias filas de una tabla o vista. Una forma simplificada de la sintaxis de DELETE es:

```
DELETE FROM tabla_o_vista WHERE condición
```

El argumento *tabla\_o\_vista* nombra la tabla o vista de la que se van a eliminar las filas. Se eliminan todas las filas de *tabla\_o\_vista* que reúnan los requisitos de la *condición* de búsqueda de la cláusula WHERE. Si no se especifica una cláusula WHERE, se eliminarán todas las filas de *tabla\_o\_vista*.

La sentencia siguiente borra todas las filas de la tabla EMP4.



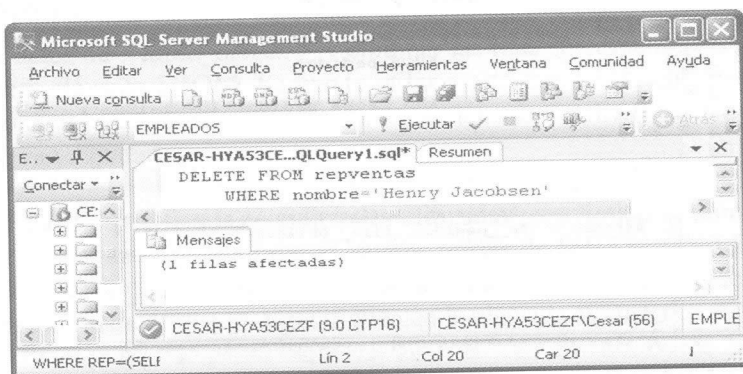
Cualquier tabla de la que se hayan quitado todas las filas sigue permaneciendo en la base de datos. La instrucción DELETE sólo elimina filas de la tabla y ésta se debe quitar de la base de datos mediante la instrucción DROP TABLE.

La siguiente consulta borra todos los vendedores de la tabla EMP con comisión inferior a 100 en el último mes.

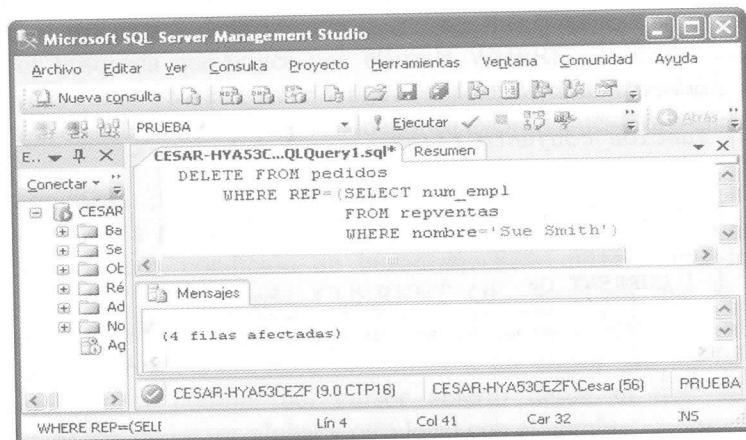


La subconsulta SELECT en la instrucción DELETE se puede utilizar para eliminar valores de más de una fila al mismo tiempo. A veces, la selección de las filas a borrar debe efectuarse en base a datos contenidos en otras tablas que se citarán dentro de la subconsulta SELECT.

A continuación se borra a Henry Jacobsen de la tabla REPVENTAS de la base de datos EMPLEADOS.

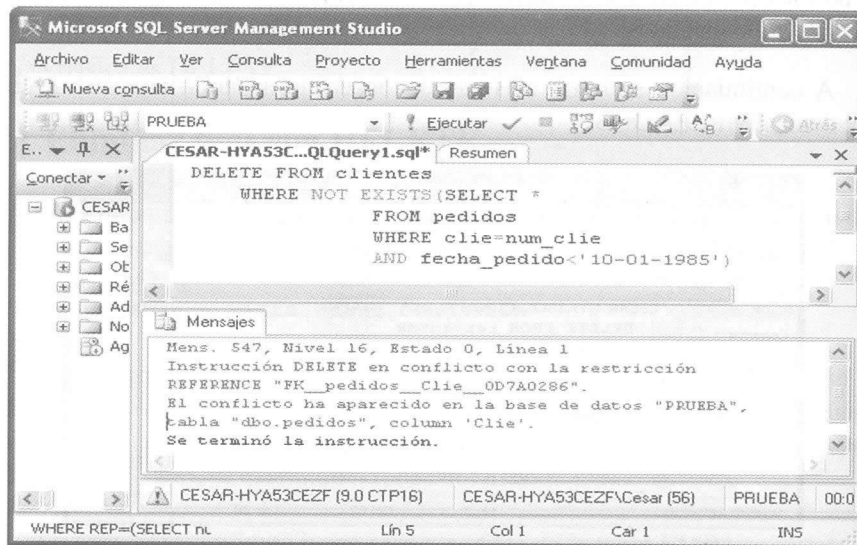


La sentencia siguiente suprime los pedidos aceptados por Sue Smith.



Las subconsultas en la cláusula WHERE de la sentencia DELETE pueden anidarse al igual que sucedía en la cláusula WHERE de la sentencia SELECT. También pueden contener referencias externas a la tabla destino de la sentencia DELETE con la finalidad de implementar la composición entre las tablas de la subconsulta y la tabla destino de la sentencia DELETE. La única restricción al uso de subconsultas en una sentencia DELETE es que la tabla destino no puede aparecer en la cláusula FROM de la subconsulta ni en ninguna de sus subconsultas a ningún nivel de anidación.

Pueden existir restricciones de integridad que impidan la eliminación de filas de una tabla tal y como se observa en el ejemplo siguiente:



Una sintaxis más general de la sentencia DELETE podría ser la siguiente:

```
DELETE
[FROM ]
{
    nombre_tabla WITH ( < sugerencia_tabla > [ ...n ] )
    | nombre_vista
    | función_conjunto_filas
}

[ FROM { < tabla_origen > } [ ,...n ] ]
[ WHERE
{
    < condición_búsqueda >
    | { [ CURRENT OF
        { { [ GLOBAL ] nombre_cursor }
          | nombre_variable_cursor
        }
      ] }
}
]
]
[ OPTION ( < sugerencia_consulta > [ ,...n ] ) ]
```

FROM es una palabra clave opcional que se puede utilizar entre la palabra clave DELETE y el destino *nombre\_tabla*, *nombre\_vista* o *función\_conjunto\_filas*. El argumento *nombre\_tabla* es el nombre de la tabla de la que se van a quitar las filas. WITH (<sugerencia\_tabla> [...n]) especifica una o más sugerencias de tabla permitidas para una tabla de destino. La palabra clave WITH y los paréntesis son obligatorios. No se permiten READPAST, NOLOCK ni READUNCOMMITTED. El argumento *nombre\_vista* es el

nombre de una vista que debe poder actualizarse y debe hacer referencia exacta a una tabla base en la cláusula FROM de la vista. Si la tabla o vista existe en otra base de datos o tiene un propietario distinto al usuario actual, utilice un nombre calificado con cuatro partes con el formato *nombre\_servidor.base\_datos.[propietario].nombre\_objeto*. El argumento *función\_conjunto\_filas* es una función OPENQUERY u OPENROWSET, según las capacidades del proveedor.

FROM <tabla\_origen> especifica una cláusula FROM adicional que permite especificar datos de <tabla\_origen> y eliminar las filas correspondientes de la tabla en la primera cláusula FROM. Se puede utilizar esta extensión, que especifica una combinación, en lugar de una subconsulta en la cláusula WHERE para identificar las filas que se van a quitar.

WHERE especifica las condiciones utilizadas para limitar el número de filas que se van a eliminar. Si no se proporciona una cláusula WHERE, DELETE quitará todas las filas de la tabla. Hay dos formas de operaciones de eliminación, que se basan en las condiciones que se especifiquen en la cláusula WHERE.

Las eliminaciones por búsqueda especifican una condición de búsqueda que califica las filas que se van a eliminar y las eliminaciones por posición utilizan la cláusula CURRENT OF para especificar un cursor. La operación de eliminación se produce en la posición actual del cursor. Este método puede ser más preciso que una eliminación DELETE por búsqueda que utilice una cláusula WHERE *condición\_búsqueda* para calificar las filas que se van a eliminar. Una eliminación DELETE por búsqueda elimina varias filas si la condición de búsqueda no identifica exclusivamente una única fila.

El argumento *condición\_búsqueda* especifica las condiciones restrictivas de las filas que se van a eliminar. No hay límite en el número de predicados que se pueden incluir en una condición de búsqueda. Para obtener más información, consulte *Condición de búsqueda*. CURRENT OF especifica que la instrucción DELETE se ejecutará en la posición actual del cursor especificado. GLOBAL especifica qué *nombre\_cursor* hace referencia a un cursor global. El argumento *nombre\_cursor* es el nombre del cursor abierto desde el que se realiza la recuperación. Si hay un cursor global y otro local con el nombre *nombre\_cursor*, este argumento hace referencia al cursor global si se especifica GLOBAL; de lo contrario, *nombre\_cursor* hace referencia al cursor local. El cursor debe permitir actualizaciones. El argumento *nombre\_variable\_cursor* es el nombre de una variable de cursor. La variable de cursor debe hacer referencia a un cursor que permita realizar actualizaciones.