

Practicar con la eliminación de registros

Cuando tenemos que eliminar filas de una tabla podemos utilizar:

- DELETE Elimina filas de una tabla asegurando la integridad referencial.
- TRUNCATE Realiza un borrado masivo de todas las filas de una tabla.

TRUNCATE se utilizará en caso especiales ya que para poder ejecutar esta operación la tabla no puede intervenir como tabla principal en una relación (no puede ser padre).

DELETE es la que se va utilizar en la mayoría de los casos.

Sólo tenemos que tener en cuenta:

- En qué tabla están las filas a borrar.
- Si queremos eliminar todas las filas o sólo algunas lo cual indicaremos mediante TOP o WHERE.
- Puede ocurrir un error cuando se ejecute, si algún borrado provoca un error de integridad referencial queriendo borrar una fila referenciada (un padre que tenga hijos).

1.- Eliminar todos los pedidos.

Las filas a eliminar se encuentran en la tabla Pedidos y las queremos eliminar todas luego no añadimos nada más.

```
DELETE Pedidos
```

O bien

```
DELETE FROM Pedidos
```

Esta FROM es opcional pero es obligatoria en muchos sistemas, mejor utilizarla.

2.- Eliminar los pedidos de hoy.

En este caso no queremos borrar todas las filas, sólo las que son de hoy, hay que añadir pues el WHERE correspondiente:

```
DELETE FROM Pedidos
WHERE fechapedido=GETDATE();
```

3.- Eliminar los pedidos del cliente Antonio Perez.

En este caso la condición del WHERE sería nombre='Antonio Perez' pero el campo nombre no está en la tabla Pedidos, luego utilizaremos una subconsulta para formular la pregunta:

```
DELETE FROM Pedidos
WHERE clie IN (SELECT numclie FROM Clientes WHERE nombre = 'Antonio Perez');
```

O bien

```
DELETE FROM Pedidos
WHERE EXISTS (SELECT * FROM Clientes
              WHERE nombre='Antonio Perez' AND clie=numclie);
```

Podríamos haber utilizado también la segunda FROM para definir un origen que incluya las dos tablas:

```
DELETE FROM Pedidos
FROM Pedidos INNER JOIN Clientes ON clie=numclie
WHERE nombre = 'Antonio Perez'
```

Pero esta forma no te la recomiendo porque es menos estándar, y es mejor acostumbrarse a escribir sentencias que sirven para cualquier sistema gestor de bases de datos.

4.- Eliminar el primer pedido.

Aquí tenemos que indicar qué pedido queremos eliminar pero con la cláusula TOP.

```
DELETE TOP (1) FROM Pedidos
```

5.- Eliminar el último pedido.

En este caso sería TOP (1) pero ordenando los pedidos por código descendente para tener el último registro introducido (el de código mayor) el primero. El TOP del DELETE, como pasa con el UPDATE no admite ORDER BY, lo podemos solucionar con una tabla derivada.

```
DELETE FROM AliasPed  
FROM (SELECT TOP (1) * FROM Pedidos ORDER BY codigo DESC) AS AliasPed;
```

En todos estos casos como borramos pedidos, y la tabla Pedidos no es principal en ninguna de la relación en las que interviene, pues no habrá problemas de integridad referencial. La tabla Pedidos contiene claves ajenas, pero en ninguna otra tabla hay una clave ajena que hace referencia a la tabla Pedidos.

Si quisiéramos borrar clientes, entonces habría que tener en cuenta que el cliente a borrar no puede tener pedidos (en la tabla Pedidos hay la clave ajena clie que hace referencia a Clientes).

Si borramos productos hay que tener en cuenta que Productos es Principal en la relación entre Productos y Pedidos, en la tabla Pedidos hay una clave ajena que hace referencia a un producto de la tabla Productos. Luego no podré borrar un producto que tenga pedidos.

La tabla Empleados es principal en varias relaciones:

- En la relación en la que interviene la clave ajena jefe de Empleados.
- En la relación en la que interviene la clave ajena rep de Pedidos
- En la relación en la que la clave ajena es repclie en Clientes
- En la relación en la que la clave ajena es dir en Oficinas.

Por lo tanto no podremos borrar un empleado si ese empleado está en cualquiera de estas situaciones:

- Es jefe de algún empleado
- Ha hecho algún pedido
- Está asignado a algún cliente
- Es director de alguna oficina.

Si queremos borrar oficinas hay que tener en cuenta que Oficinas es principal en la relación cuya clave ajena es oficina en la tabla Empleados, por lo que no podemos eliminar una oficina que esté asignada a algún empleado.