

Eliminación de registros con claves foráneas

Una de las principales bondades de las claves foráneas es que permiten eliminar y actualizar registros en cascada.

Con las restricciones de clave foránea podemos eliminar un registro de la tabla cliente y a la vez eliminar un registro de la tabla venta usando sólo una sentencia DELETE. Esto es llamado eliminación en cascada, en donde todos los registros relacionados son eliminados de acuerdo a las relaciones de clave foránea. Una alternativa es no eliminar los registros relacionados, y poner el valor de la clave foránea a NULL (asumiendo que el campo puede tener un valor nulo). En nuestro caso, no podemos poner el valor de nuestra clave foránea id_cliente en la tabla venta, ya que se ha definido como NOT NULL. Las opciones estándar cuando se elimina un registro con clave foránea son:

.	ON DELETE RESTRICT
.	ON DELETE NO ACTION
.	ON DELETE SET DEFAULT
.	ON DELETE CASCADE
.	ON DELETE SET NULL

ON DELETE RESTRICT es la acción predeterminada, y no permite una eliminación si existe un registro asociado, como se mostró en el ejemplo anterior. ON DELETE NO ACTION hace lo mismo.

ON DELETE SET DEFAULT actualmente no funciona en MySQL - se supone que pone el valor de la clave foránea al valor por omisión (DEFAULT) que se definió al momento de crear la tabla.

Si se especifica ON DELETE CASCADE, y una fila en la tabla padre es eliminada, entonces se eliminarán las filas de la tabla hijo cuya clave foránea sea igual al valor de la clave referenciada en la tabla padre. Esta acción siempre ha estado disponible en MySQL.

Si se especifica ON DELETE SET NULL, las filas en la tabla hijo son actualizadas automáticamente poniendo en las columnas de la clave foránea el valor NULL. Si se especifica una acción SET NULL, debemos asegurarnos de no declarar las columnas en la tabla como NOT NULL.

A continuación se muestra un ejemplo de eliminación en cascada:

```
mysql> ALTER TABLE venta ADD FOREIGN KEY(id_cliente)
-> REFERENCES cliente(id_cliente) ON DELETE CASCADE;
Query OK, 3 rows affected (0.23 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

Vamos a ver como están nuestros registros antes de ejecutar la sentencia DELETE:

```
mysql> SELECT * FROM cliente;
```

```
+-----+-----+
```

```

| id_cliente | nombre      |
+-----+
|          1 | Juan Penas  |
|          2 | Pepe el toro |
|          3 | Pepe pecas  |
+-----+
3 rows in set (0.00 sec)

```

```

mysql> SELECT * FROM venta;
+-----+-----+-----+
| id_factura | id_cliente | cantidad |
+-----+-----+-----+
|          1 |          1 |        23 |
|          2 |          3 |        39 |
|          3 |          2 |        81 |
+-----+-----+-----+
3 rows in set (0.00 sec)

```

Ahora eliminaremos a Pepe Pecas de la base de datos:

```

mysql> DELETE FROM cliente WHERE id_cliente=3;
Query OK, 1 row affected (0.05 sec)

```

```

mysql> SELECT * FROM venta;
+-----+-----+-----+
| id_factura | id_cliente | cantidad |
+-----+-----+-----+
|          1 |          1 |        23 |
|          3 |          2 |        81 |
+-----+-----+-----+
2 rows in set (0.00 sec)

```

```

mysql> SELECT * FROM cliente;
+-----+-----+
| id_cliente | nombre      |
+-----+-----+
|          1 | Juan Penas  |
|          2 | Pepe el toro |
+-----+-----+
2 rows in set (0.00 sec)

```

Con la eliminación en cascada, se ha eliminado el registro de la tabla venta al que estaba relacionado Pepe Pecas.