¿Qué es el backup de MySQL?

Este tipo de **backup** realiza una copia de los archivos de datos reales de la base de datos. Por ejemplo, podemos usar mysqlbackup para bases de datos InnoDB y mysqlhotcopy para MyISAM. Este tipo de **backup** permite restaurar una copia de la base de datos mucho más rápido que el **backup** lógico.

BACKUP: INNODB

En el almacenamiento de los datos en MySQL, InnoDB desempeña un papel importante. InnoDB es un subsistema de almacenamiento que ha prevalecido sobre otros motores de almacenamiento en los últimos años, sobre todo gracias a su mayor rendimiento y fiabilidad.

Su gran fiabilidad, incluso a muy alto rendimiento, ha llevado a MySQL a hacer de InnoDB el motor de almacenamiento predeterminado en las versiones 5.6 y posteriores.

MySQL es una base de datos de código abierto que permite el uso de diferentes motores de almacenamiento. Esto ofrece mucha flexibilidad a las bases de datos para el almacenamiento web. Además de InnoDB y su predecesor oficial, MyISAM, existen otros motores, como BerkeleyDB, CSV, NDB o Federated Engine.

Las transacciones son paquetes de datos que consisten en varias operaciones que no se pueden separar. Si no se puede ejecutar una operación, toda la transacción se cancela. Para solucionarlo, se deben deshacer todas las operaciones anteriores, no solo la errónea. Este proceso lleva más tiempo que una operación simple, pero solo así se puede garantizar la coherencia de la base de datos.

El motor de almacenamiento InnoDB no siempre es el mejor sustituto de su predecesor, MyISAM. En cuanto los proyectos o la base de datos crecen en complejidad, InnoDB pasa a ser la solución apropiada para MySQL. Para bases de datos o aplicaciones más pequeñas, MyISAM es mucho más potente. Si se usan correctamente, no hay cuestión sobre cuál de los motores de almacenamiento se debe usar. Si es necesario, en la misma base de datos se pueden usar ambos módulos, uno junto al otro.

Triggers Mysql

Un trigger, también conocido como disparador (Por su traducción al español) es un conjunto de sentencias SQL las cuales se ejecutan de forma automática cuando ocurre algún evento que modifique a una tabla. Pero no me refierón a una modificación de estructura, no, me refiero a una modificación en cuando a los datos almacenados, es decir, cuando se ejecute una sentencia INSERT, UPDATE o DELETE.

A diferencia de una función o un store procedure, un trigger no puede existir sin una tabla asociada.

Lo interesante aquí es que podemos programar los triggers de tal manera que se ejecuten antes o después, de dichas sentencias; Dando como resultado seis combinaciones de eventos.

BEFORE INSERT Acciones a realizar antes de insertar uno más o registros en una tabla.

AFTER INSERT Acciones a realizar después de insertar uno más o registros en una tabla.

BEFORE UPDATE Acciones a realizar antes de actualizar uno más o registros en una tabla.

AFTER UPDATE Acciones a realizar después de actualizar uno más o registros en una tabla.

BEFORE DELETE Acciones a realizar antes de eliminar uno más o registros en una tabla.

AFTER DELETE Acciones a realizar después de eliminar uno más o registros en una tabla.

A partir de la versión 5.7.2 de MySQL podemos tener la n cantidad de triggers asociados a una tabla. Anteriormente estábamos limitados a tener un máximo de seis trigger por tabla (Uno por cada combinación evento).

Podemos ver esto como una relación uno a muchos, una tabla puede poseer muchos triggers y un trigger le pertenece única y exclusivamente a una tabla.

**Clientes**:

idCliente, Nombre, Dirección, Teléfono, Correo, Fecha\_Nacimiento, Sexo.

Clientes -> compran productos

1 Cliente puede hacer muchas compras

**Productos**:

IdProducto, titulo, descripción, imagen, precio, stock, idCategoria.

Pertenecen a una categoria especifica.

Un producto puede estar en muchas compras.

**Categorías**:

IdCategoria, categoría, descripción.

1 categoria puede estar asociada a muchos productos.

**Compra:**

idCompra, codigo, fecha\_venta, monto\_final, idCliente.

Ventas están asociadas a la tabla detalle\_venta

1 venta puede estar relacionada a varios detalles.

**Detalle Ventas**:

IdDetalle, cantidad, precio, idProducto, idCompra.