Transacciones

Sitio: Agencia de Aprendizaje a lo largo de la Vida
Curso: Administración de Base de Datos 1° F
Libro: Transacciones

Imprimido por: MARIO DAVID GONZALEZ BENITEZ
Día: domingo, 20 de octubre de 2024, 22:46

Tabla de contenidos

Introducción

¿Qué significa ULT?

Se denominan unidad lógica de trabajo (ULT) a una secuencia de operaciones que modifican los datos pasándolos de un estado coherente a otro.

Es una secuencia de actualizaciones sobre las tablas que debe realizarse completa o no realizarse en absoluto, pero nunca a medias, porque si así fuera los datos podrían quedar lógicamente incoherentes.

¿Cuál es su funcionalidad?

La decisión de cuales sentencias de actualización deben incluirse en una ULT depende de la lógica de las aplicaciones, por lo que son éstas las que deben decírselo al sistema gestor, a cambio, éste se responsabiliza de que las sentencias incluidas en una ULT se ejecuten todas o ninguna.

En principio el sistema gestor considera que todas las sentencias de actualización ejecutadas dentro de un programa forman una ULT.

Puede ocurrir que varios programas se ejecuten concurrentemente, esto significa que en un momento dado puede haber varias ULT, es decir, varias secuencias de actualizaciones en progreso a la vez.

Importante

Para prevenir y evitar esta situación los sistemas gestores suelen emplear una técnica consistente en bloquear todos los datos modificados por una ULT, conforme se vayan ejecutando sus sentencias de actualización. Esto significa que estos datos son por un momento inaccesibles para las restantes ULT. Si alguna ULT necesitara acceder a uno de ellos, el sistema gestor la hará esperar hasta que la primera ULT termine.

Se debe evitar ULT que duren mucho tiempo ya que puede provocar esperas largas en otras ULT.

Transacciones: reglas generales

¿Cuáles son?

Por regla general en sistemas gestores relacionales modernos disponemos de tres tipos de transacciones según la forma de iniciarlas:

- De confirmación automática : el gestor de datos inicia una transacción automáticamente por cada operación que actualice datos. De este modo mantiene siempre la consistencia de la base de datos, aunque puede generar bloqueos.
- Implícitas: cuando el gestor de datos comienza una transacción automáticamente cada vez que se produce una actualización de datos, pero el que dicha transacción se confirme o se deshaga, lo debe indicar el programador.
- Explícitas: son las que iniciamos nosotros "a mano" mediante instrucciones SQL. Somos nosotros, los programadores, los que indicamos qué operaciones van a abarcar.

Una transacción explícita se define de manera general con una instrucción que marca su inicio, y dos posibles instrucciones que marcan su final en función de si debe tener éxito o debe fracasar en bloque.

Transacciones: MySql

MySQL es un gestor de datos relacional que soporta diversos motores de almacenamiento, solamente InnoDB soporta el uso de transacciones.

- *Start Transaction o Begin*: marca el inicio de una transacción. Se suele usar más a menudo *BEGIN* porque es más corto.
- *Rollback:* fuerza que se deshaga la transacción en caso de haber un problema o querer abandonarla. Cierra la transacción.
- *Commit:* confirma el conjunto de operaciones convirtiendo los datos en definitivos. Marca el éxito de la operación de bloque y cierra la transacción.

Para evitar estas situaciones hay que descomponer la ULT asociada a un programa en varias más cortas. Esto se realiza mediante las sentencias *Commit* y *Rollback*.

Transacciones

Veamos un ejemplo

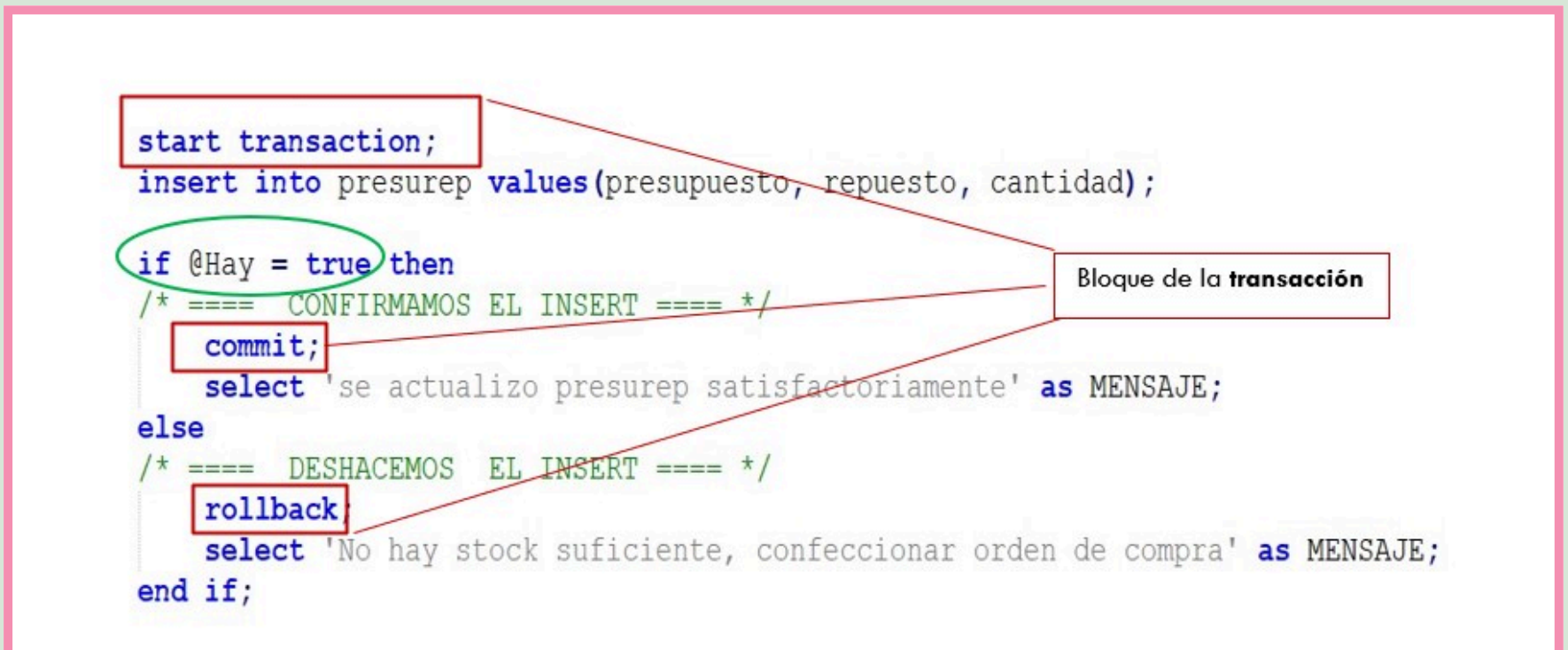
Cuando mostramos el ejemplo 1 de *trigger* en el capítulo anterior, consideramos un supuesto que es: siempre hay stock suficiente, pero en la realidad puede no cumplirse.

Tomando ese ejemplo vamos a incorporar la metodología de las transacciones.

Primero una breve aclaración en un extracto del código y luego lee la resolución en el archivo Semana12-EjemploTransaccion1.sql.

Encontrá el archivo para descargar en el materia de consulta.

Observemos la siguiente imagen de ejemplo



Resaltado en rojo se muestra el bloque de la transacción, estas palabras son claves.

El inicio lo indica *start transaction* en este caso lo colocamos antes de la ejecución de *insert*, el sistema gestor queda atento porque sabe que todo lo que ejecute después puede o no quedar registrado.

El *insert* actualiza a la base ya que agrega una fila, si se ejecuta el *commit* el sistema gestor se relaja y olvida lo ejecutado, pero si se ejecuta el *rollback* debe dejar la base en el estado en el que se encontraba al momento del *start transaction*.

Ahora podés ir al enlace para descargar el archivo y leer la resolución.