Universidad de Costa Rica

Recinto de Turrialba

Administración de bases de datos.

Mario Quirós Luna – B76090

**Tarea Moral:**

Common Table Expresion (CTE)

Una expresión de tabla común es un conjunto de resultados con nombre temporal al que se puede hacer referencia dentro de una instrucción SELECT, INSERT, UPDATE o DELETE, esta expresión también puede ser utilizada en una vista.

Sintaxis:

La consulta CTE comienza con un “WITH” y es seguida por el Nombre de expresión. Utilizaremos este nombre de expresión en nuestra consulta de selección para mostrar el resultado de nuestra consulta CTE y escribiremos nuestra definición de consulta CTE.

WITH expression\_name [(column\_name [,...n])]

AS (CTE\_query\_definition)

Para ver el resultado CTE usamos una consulta con un SELECT con el nombre de la expresión CTE.

Select [Column1, Column2, Column3…] from expression\_name

O

Select \* from expression\_name

Tipos de CTE:

CTE no recursivos

Los CTE no recursivos son simples, cuando el CTE no utiliza ninguna recursividad o procesamiento repetido de una subrutina. Crearemos un CTE simple no recursivo para mostrar el número de fila del 1 al 10.

Según la sintaxis de CTE, cada consulta de CTE comenzará con un “WITH” seguido del nombre de Expresión de CTE con la lista de columnas.

Aquí nosotros hemos estado usando solo una columna como ROWNO. La siguiente es la parte de Consulta, aquí nosotros escribimos nuestra consulta de selección para que se ejecute para nuestro CTE. Después de crear nuestra consulta CTE para ejecutar el CTE, use la instrucción select con el nombre de Expresión CTE.

WITH ROWCTE(ROWNO) AS

(

SELECT

ROW\_NUMBER() OVER (ORDER BY name ASC) AS ROWNO

FROM sys.databases

WHERE database\_id <= 10

)

SELECT \* FROM ROWCTE

Table

Description automatically generated

CTE recursivos

Los CTE recursivos son ciclos o bucles de procedimiento repetidos, también conocidos como recursividad. La consulta recursiva se llama a sí misma hasta que la consulta satisfaga la condición. En un CTE recursivo, nosotros debemos proporcionar una condición where para terminar la recursión.

Veremos cómo crear una consulta recursiva simple para mostrar el número de fila del 1 al 10 usando un CTE.

En primer lugar, declaramos la variable Integer como ”RowNo” y establecemos el valor predeterminado en 1 y entonces hemos creado nuestra primera consulta CTE como un nombre de expresión, “ROWCTE”. En nuestro CTE, primero mostraremos por defecto el número de fila predeterminado y luego usaremos Union ALL para incrementar y mostrar el número de fila 1 en uno hasta que el número de fila alcance el valor incrementado a 10. Para ver el resultado, usaremos una consulta de selección para mostrar nuestro resultado CTE.

Declare @RowNo int = 1;

;WITH ROWCTE as

(

SELECT @RowNo as ROWNO

UNION ALL

SELECT ROWNO+1

FROM ROWCTE

WHERE RowNo < 10

)

SELECT \* FROM ROWCTE

Table

Description automatically generated

Seguridad y control de acceso:

Trigger:

Un Trigger o disparador es un script que se usa en lenguaje de programación SQL, en especial en bases de datos como MySQL o PostgreSQL.

Consiste en una serie de reglas predefinidas que se asocian a una tabla. Estas reglas se aplican a la base de datos cuando se realizan determinadas operaciones en la tabla, por ejemplo, al añadir, actualizar o eliminar registros.

Dicho de otra manera, el Trigger desencadena determinadas acciones de forma automática en las tablas de la base de datos cuando se insertan, modifican y se añaden nuevos datos.

Estos disparadores se llevan usando en MySQL desde la versión 5.0.2., mientras que PostgreSQL ya los incluyó en el año 1997.

¿Para qué sirve?

La principal función de los Trigger es contribuir a mejorar la gestión de la base de datos. Gracias a ellos muchas operaciones se pueden realizar de forma automática, sin necesidad de intervención humana, lo que permite ahorrar mucho tiempo.

Otra de sus funciones es aumentar la seguridad e integridad de la información. Esto lo consiguen gracias a la programación de restricciones o requerimientos de verificación que permiten minimizar los errores y sincronizar la información.

Por otra parte, entre sus principales ventajas es que todas estas funciones se pueden realizar desde la propia base de datos, es decir, no es necesario recurrir a lenguajes externos de programación.

¿Cuándo se puede usar un Trigger?

Los Trigger se puede ejecutar cuando el usuario realizar alguna acción relacionada con añadir, actualizar o eliminar información de una tabla. Es decir, al usar los comandos INSERT, UPDATE o DELETE.

Por tanto, para poder usar un Trigger es necesario que el usuario posea permisos INSERT y DELETE e dicha base de datos.

Estructura de un Trigger

La estructura y modo de funcionamiento de un Trigger se puede resumir en tres pasos:

Se produce una llamada de activación al código que se ha de ejecutar.

Aplica las restricciones necesarias para poder realizar la acción, por ejemplo, una determinada condición o una nulidad.

Una vez verificadas las restricciones, se ejecuta la acción, en base a las instrucciones recibidas en el primer punto.

Tipos de Trigger

Existen diferentes tipos de disparadores, en función de las ejecuciones que realizan.

Disparadores de fila: también llamados row triggers, son aquellos cuya ejecución se realiza a través de llamadas desde una tabla asociada al trigger.

Disparadores de secuencia: también llamados statement triggers, son aquellos que se ejecutan solo una vez, independientemente de la cantidad de veces que se cumplan las condiciones para su ejecución.

¿Cuándo ejecuta su acción un Trigger?

Los triggers pueden ejecutar su acción en diferentes momentos.

* Antes de ejecutar la sentencia (before statement)
* Después de ejecutar la sentencia y de comprobar las restricciones y condiciones aplicables (after statement)
* Antes de modificar la fila de la tabla afectada por la sentencia del trigger, y de comprobar las restricciones y condiciones de ejecución (before row)
* Después de modificar la fila de la tabla afectada por la sentencia del trigger, y de comprobar las restricciones y condiciones de ejecución (after row)