

Laboratorio # 4

• Triggers



Trabajando bajo el concepto de Modularización como se vio en la práctica pasada, existe otro tipo de subprograma que ayuda a obtener una mejor manera de administrar o manejar los datos de las tablas de una base de datos, en este caso corresponde hablar acerca de los: **Triggers**.

¿Qué son los Triggers?

Los Triggers (Disparadores) son utilizados para poder definir una acción que nuestra base de datos debería siempre realizar cuando ocurre algún tipo de cambio que pueda afectar alguna de las tablas.

Dichos Triggers están contruidos por bloques PL/SQL, los cuales se ejecutan al momento de recibir una determinada instrucción SQL, en este caso hablamos de las instrucciones **DML** (**INSERT**, **UPDATE**, **DELETE**) hacia una tabla.

Un dato importante para la creación de Triggers en ORACLE es que no podemos crear y usar Triggers con el usuario SYS, por regla de ORACLE, por dicha regla se debe crear un nuevo usuario. Antes de crear un nuevo usuario, se debe crear un **TABLESPACE** que permitirá el almacenamiento general de datos.

Sintaxis para crear un TABLESPACE

```
CREATE TABLESPACE nombre_del_tablespace  
    DATAFILE 'nombre_del_tablespace.df'  
    SIZE [Tamaño en M] AUTOEXTENDED ON;
```

Sintaxis para crear un usuario

```
CREATE USER nombre_del_usuario  
    IDENTIFIED BY su_contraseña  
    DEFAULT TABLESPACE nombre_del_tablespace  
    TEMPORARY TABLESPACE temp  
    QUOTA UNLIMITED ON nombre_del_tablespace;
```

Si solo se crea el usuario, de entrada, este no podrá tener acceso a varias de las operaciones que son de suma importancia en una base de datos, por lo cual se le deben otorgar privilegios al usuario que le permitan poder realizar las operaciones que el guste, en esta práctica los privilegios que deben tener los usuarios creados son los siguientes:

```
GRANT CONNECT, RESOURCE, DBA TO nombre_del_usuario;
GRANT CREATE SESSION TO nombre_del_usuario;
```

Para poder definir un Trigger se debe tener en cuenta lo siguiente:

- Especificar las condiciones bajo las cuales el Trigger será ejecutado.
- Especificar las acciones que se realizarán para cuando el Trigger deba ser ejecutado.

Sintaxis de un Trigger

Con lo anteriormente expuesto, se tiene entonces la siguiente estructura para crear un Trigger dentro de la base de datos:

```
CREATE [OR REPLACE] TRIGGER <nombre_del_trigger>
[BEFORE|AFTER|INSTEAD OF] --Momento en el que se ejecutara el Trigger
[INSERT|DELETE|UPDATE] --Evento por el cual será disparado el Trigger
ON <nombre_de_tabla> o <nombre_de_la_vista>
FOR EACH [ROW|STATEMENT] [WHEN <condiciones>]
    --Bloque PL/SQL
DECLARE
    --Declaración de variables
BEGIN
    --Instrucciones de las acciones
END;
```

Si el evento que se desea seleccionar es el **UPDATE** se debe especificar los campos de la tabla a los cuales se desea hacer dicha actualización, de la siguiente forma:

```
UPDATE [OF <col1>, ..., <colN>]
```

En la siguiente lista se detallan cada uno de los componentes y sus diferentes funciones dentro de la creación del Trigger:

BEFORE/AFTER/INSTEAD OF: Indica el instante de ejecución del Trigger con respecto al evento que lo invocara.

Evento (Instrucción DML): Es aquella instrucción DML que se realiza en una tabla que provoca que el Trigger se active.

Niveles:

FOR EACH STATEMENT: Son Triggers a nivel de orden (a nivel de sentencias), solo se activarán una vez.

FOR EACH ROW: Son Triggers a nivel de fila, lo cual indica que el Trigger se activara cada vez que una fila de una tabla se vea afectada.

WHEN: Condición que sirve para evaluar si algo es verdadero o falso (se usa con FOR EACH ROW).

REGISTROS :OLD Y :NEW: Para referirse al valor nuevo y al antiguo de una columna de una fila de la tabla, se utilizan los prefijos :OLD y :NEW.

En este esquema se puede ver cómo trabajan los prefijos a fondo:

Suceso	:old	:new
INSERT	NULL	Nuevos Valores
UPDATE	Antiguos Valores	Nuevos Valores
DELETE	Antiguos Valores	NULL

Existen formas de poder habilitar o deshabilitar Triggers, la sintaxis para poder activar o desactivar un Trigger es la siguiente:

```
--Para un Trigger en específico
ALTER TRIGGER nombre_del_trigger [ENABLE|DISABLE];

--Para todos los Triggers
ALTER TABLE nombre_de_la_tabla [ENABLE|DISABLE] ALL TRIGGERS;
```

Para eliminar un Trigger se tiene la siguiente estructura:

```
DROP TRIGGER nombre_del_trigger;
```