Triggers PL/SQL

Laboratorio de Base de datos

Cristian E. Sánchez

Triggers PL/SQL

- Los triggers son bloques pl/sql almacenados en la base de datos Oracle y se **ejecutan automáticamente** cuando se dispara un evento.
- Los eventos pueden ser cualquiera de los siguientes:
  - Una sentencia DML (data manipulation language) ejecutada contra una tabla e.g, **Insert, Update,** o **Delete.** Por ejemplo, si defines un trigger que se ejecuta antes de un **Insert** en una tabla **Clientes,** entonces el trigger ejecuta una acción justo antes de que un nuevo registro se haya insertado en la tabla **Clientes.**
  - Una sentencia DDL (data manipulation language) ejecutada e.g., **Create** o **Alter.** Generalmente son usados para tener auditoría de los cambios en el esquema.

UNIVERSIDAD ICESI

Principales usos

- Prevenir transacciones inválidas.
- Recolectar información estadística sobre accesos a tablas.
- Generar valores valores automáticamente.
- Auditar datos sensibles.

UNIVERSIDAD ICESI

Creación de un Trigger PL / SQL

```
CREATE [OR REPLACE] TRIGGER trigger_name
{BEFORE | AFTER } triggering_event ON table_name
[FOR EACH ROW]
[FOLLOWS | PRECEDES another_trigger]
[ENABLE / DISABLE ]
[WHEN condition]
DECLARE
    declaration statements
BEGIN
    executable statements
EXCEPTION
    exception_handling statements
END;
```

UNIVERSIDAD ICESI

Sintaxis de la creación del trigger

Encabezado

```
CREATE [OR REPLACE] TRIGGER trigger_name
    {BEFORE | AFTER } triggering_event ON table_name
    [FOR EACH ROW]
    [FOLLOWS | PRECEDES another_trigger]
    [ENABLE / DISABLE ]
    [WHEN condition]
```

Cuerpo

```
DECLARE
    declaration statements
BEGIN
    executable statements
EXCEPTION
    exception_handling statements
END;
```

UNIVERSIDAD ICESI

CREATE OR REPLACE

- La palabra clave CREATE indica que está creando un nuevo Trigger. Las palabras clave OR REPLACE son opcionales. Se utilizan para modificar un disparador existente.
- Aunque las palabras clave OR REPLACE son opcionales, aparecen con la palabra clave CREATE en la mayoría de los casos.
- Supongamos que define un nuevo trigger llamado **trigger_example**

```
CREATE TRIGGER trigger_example
    ...
```

- Si luego decide modificarlo y no se incluye la palabra OR REPLACE, entonces se genera un error indicando que el trigger ya existe.

UNIVERSIDAD ICESI

CREATE OR REPLACE

- De esta manera, las palabra clave CREATE OR REPLACE reemplazan el trigger existente y crea uno nuevo.

```
CREATE OR REPLACE trigger_example
    ....
```

UNIVERSIDAD ICESI

Nombre del trigger

- Indica el nombre del trigger que se desea crear después de las palabra clave CREATE OR REPLACE

```
CREATE OR REPLACE trigger_example
    ....
```

UNIVERSIDAD ICESI

BEFORE | AFTER

- Estas opciones indican cuando se dispara el trigger, ya sea antes o después del evento disparador e.g, **INSERT, UPDATE** o **DELETE**

```
CREATE [OR REPLACE] TRIGGER trigger_name
    {BEFORE | AFTER } triggering_event ON table_name
    [FOR EACH ROW]
    [FOLLOWS | PRECEDES another_trigger]
    [ENABLE / DISABLE ]
    [WHEN condition]
```

UNIVERSIDAD ICESI

FOR EACH ROW

- Esta cláusula indica que el trigger row-level. Es decir, el trigger se dispara una vez por cada fila insertada, actualizada o eliminada.

```
CREATE [OR REPLACE] TRIGGER trigger_name
    {BEFORE | AFTER } triggering_event ON table_name
    [FOR EACH ROW]
    [FOLLOWS | PRECEDES another_trigger]
    [ENABLE / DISABLE ]
    [WHEN condition]
```

UNIVERSIDAD ICESI

ENABLE / DISABLE

- Esta opción especifica si el trigger es creado en el estado **habilitado** o **deshabilitado.**
- Por defecto, si no se especifica el estado entonces el trigger es creado como **enabled.**

```
CREATE [OR REPLACE] TRIGGER trigger_name
    {BEFORE | AFTER } triggering_event ON table_name
    [FOR EACH ROW]
    [FOLLOWS | PRECEDES another_trigger]
    [ENABLE / DISABLE ]
    [WHEN condition]
```

UNIVERSIDAD ICESI

FOLLOWS | PRECEDES another_trigger

- Por cada evento disparador e.g, **INSERT, UPDATE,** o **DELETE,** se pueden definir multiples triggers a disparar. En este caso, se debe especificar la secuencia usando las opciones FOLLOWS o PROCEDES.

```
CREATE [OR REPLACE] TRIGGER trigger_name
    {BEFORE | AFTER } triggering_event ON table_name
    [FOR EACH ROW]
    [FOLLOWS | PRECEDES another_trigger]
    [ENABLE / DISABLE ]
    [WHEN condition]
```

UNIVERSIDAD ICESI

Ejemplo de un trigger – Oracle

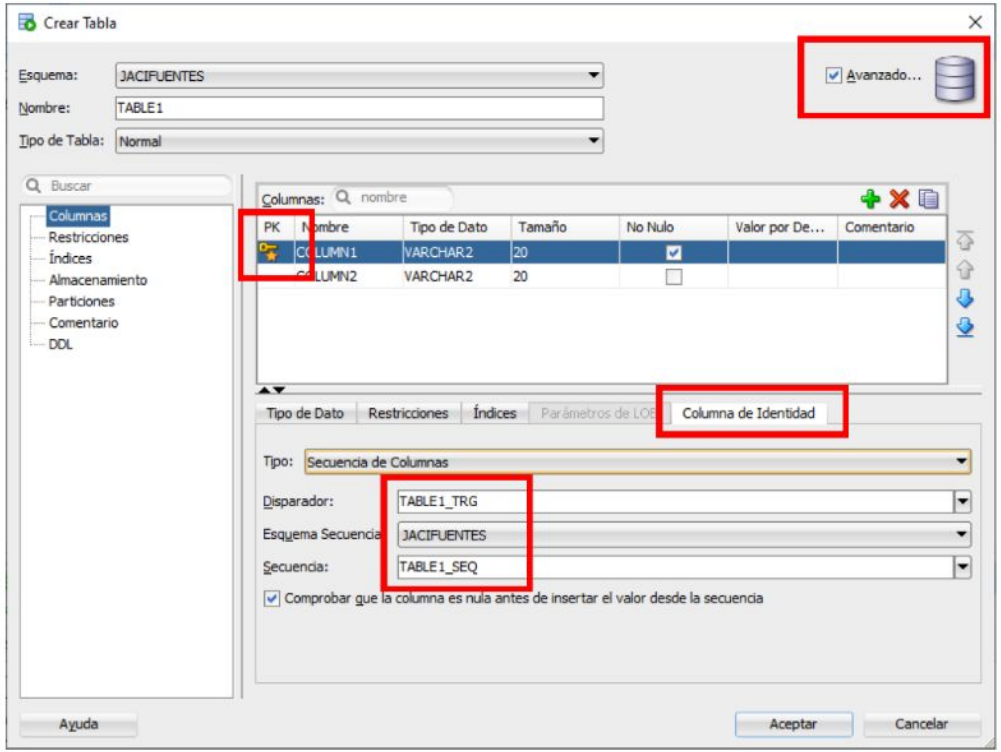
- Suponga que queremos almacenar las acciones realizadas sobre la tabla Estudiantes cada vez que un estudiantes es actualizado o eliminado.
- Creamos una nueva tabla para auditar los eventos UPDATE y DELETE

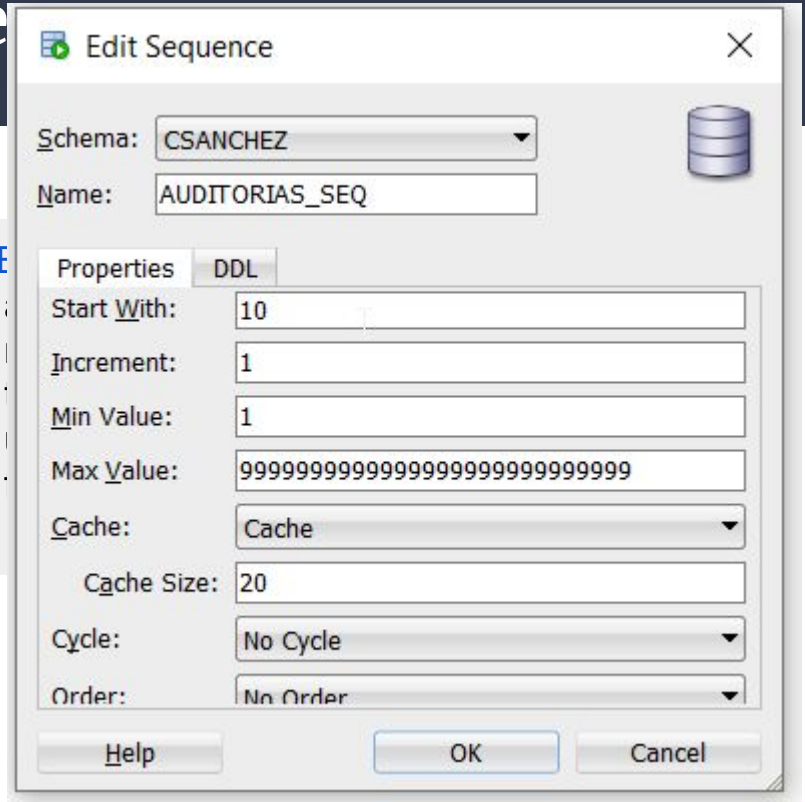
```
CREATE TABLE auditorias (
    auditoria_id NUMBER PRIMARY KEY,
    nombre_tabla VARCHAR2(255),
    transaccion VARCHAR2(10),
    usuario VARCHAR2(30),
    fecha_transaccion DATE
);
```

UNIVERSIDAD ICESI

Ejemplo de un trigger – Oracle

(*) Para generar los valores automáticamente, tener en cuenta lo siguiente:





UNIVERSIDAD ICESI

Ejemplo de un trigger – Oracle

- Ahora, creamos un nuevo trigger asociado a la tabla **Estudiantes**

```
1  CREATE OR REPLACE TRIGGER trg_auditoria_estudiantes
2      AFTER
3      UPDATE OR DELETE
4      ON estudiantes
5      FOR EACH ROW
6  DECLARE
7     tipo_transaccion VARCHAR2(10);
8  BEGIN
9     -- determinamos el tipo de transacción
10    tipo_transaccion := CASE
11          WHEN UPDATING THEN 'UPDATE'
12          WHEN DELETING THEN 'DELETE'
13    END;
14
15    -- insertar registro en la tabla de auditoria
16    INSERT INTO auditorias (nombre_tabla , transaccion , usuario, fecha_transaccion)
17    VALUES('ESTUDIANTES', tipo_transaccion, USER, SYSDATE);
18 END;
```

UNIVERSIDAD ICESI

Ejemplo de un trigger – Oracle

- Ahora suponga que el área financiera de la universidad Icesi tiene una regla la cual indica que el valor del concepto de una factura no puede ser cambiada por más del 20% del valor original.
- Crea un trigger denominado ‘valorconcepto_revisor’ para forzar esta regla. El trigger se dispara cada vez que haya una actualización sobre la tabla **Facturas** y genera un mensaje de error cuando dicha regla es violada.

UNIVERSIDAD ICESI

Ejemplo de un trigger – Oracle

```
1  create or replace trigger valorconcepto_revisor
2  before update on facturas
3  for each row
4  begin
5  if ((:new.valor_concepto/:old.valor_concepto) >= 1.2) or
6  ((:old.valor_concepto/:new.valor_concepto) >= 1.2)
7  then
8  RAISE_APPLICATION_ERROR(-20002, 'Warning: Ese cambio no está permitido.');
9  end if;
10 end;
```

UNIVERSIDAD ICESI