

# BBDD

## Bases de Datos – Programación de bases de datos II



# Objetivos

- Identificar y diferenciar los distintos tipos de encapsulamiento de código y su aplicación: bloques anónimos, procedimientos almacenados, funciones de usuario, eventos, Triggers.
- Desarrollar los distintos tipos de eventos, evaluando y utilizando las sentencias del lenguaje incorporado en el SGBD.
- Desarrollar los distintos tipos de disparadores, evaluando y utilizando las sentencias del lenguaje incorporado en el SGBD.
- Utilizar los recursos de programación en las encapsulaciones de código: cursores, transacciones y la gestión de excepciones.



# Cursores

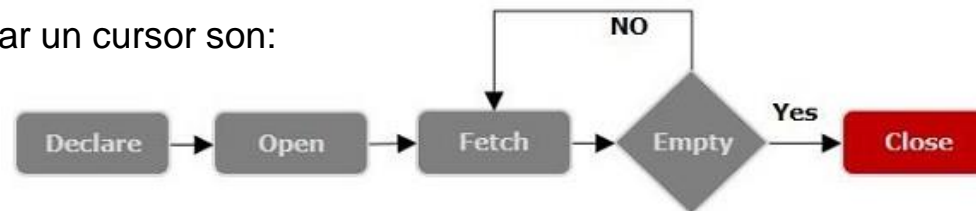
- Los cursores son recursos de programación que nos permiten recorrer y posteriormente procesar **uno a uno** (secuencialmente) los registros devueltos por el SGBD para una determinada consulta de **selección**.
- Se usan en nuestro código de programación de bases de datos cuando se precisa realizar una lógica compleja fila a fila.
- Los campos que recuperemos en la consulta de selección del cursor se recogen cada uno en una variable.
- Podemos considerarlo como un **iterador** sobre la colección de filas que nos devolverá la consulta.
- Los cursores tienen las siguientes propiedades:

**Asensitive:** Las modificaciones realizadas sobre la tabla consultada una vez creado el cursor, NO se reflejan en los datos recuperados por el cursor.

**Read only:** Únicamente de lectura. No permiten actualizar los datos de la tabla consultada.

**Nonscrollable:** Sólo pueden ser recorridos en una dirección y no podemos saltarnos filas ni cambiar de dirección.

- Los pasos para utilizar un cursor son:



- La declaración de un cursor debe aparecer antes de las declaraciones de los manejadores de errores (HANDLER) y después de la declaración de variables locales.
- Cuando el CURSOR ya no dispone de más registros, se dispara un error (NOT FOUND) que debemos gestionar con un HANDLER.

- Ejemplo de cursor en un procedimiento almacenado:

```
DELIMITER $$
CREATE PROCEDURE procedimiento()
BEGIN
    DECLARE done BOOLEAN DEFAULT FALSE;
    DECLARE a, b SMALLINT;
    DECLARE curl CURSOR FOR SELECT id, data FROM tabla;
    DECLARE CONTINUE HANDLER FOR NOT FOUND
    BEGIN
        SET done = TRUE;
    END;

    OPEN curl;

    WHILE done = FALSE DO
        FETCH curl INTO a, b;
        IF done = FALSE THEN
            IF a < b THEN
                INSERT INTO tabla2 VALUES (a,b);
            END IF;
        END IF;
    END WHILE;

    CLOSE curl;
END $$
DELIMITER ;
```