

M10UF1 - LENGUATGES SQL: DCL I EXTENSIÓ PROCEDIMENTAL. SGBD CORPORATIU



ÍNDICE

1. CURSORES.....	2
2. HANDLERS.....	3
3. VIEWS.....	4
4. USUARIOS.....	4

1. CURSORES

Los cursores en SQL son objetos utilizados para manipular conjuntos de resultados de consultas una consulta fila por fila y realizar operaciones específicas en cada una de ellas.

Se siguen algunos pasos para utilizar un cursor:

- **Declaración del cursor:** Se define la consulta y los parámetros necesarios.
- **Apertura del cursor:** Se ejecuta la consulta y se almacenan los resultados en el cursor.
- **Recorrido del cursor:** Se avanza fila por fila para realizar operaciones en cada una de ellas.
- **Manipulación de datos:** Durante el recorrido del cursor, se pueden realizar diferentes operaciones en las filas individuales.
- **Cierre del cursor:** Se cierra el cursor para liberar recursos y resultados de la consulta.

```
DECLARE cur CURSOR FOR SELECT id_character, hp FROM stats;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET salir = 1;

OPEN cur;

bucle:LOOP
    FETCH cur INTO id_char, hp_char;

    IF salir = 1 THEN
        SELECT CONCAT ("Fin del bucle") AS end_text;
        LEAVE bucle;
    END IF;

    IF hp_char <= 26 THEN
        UPDATE stats SET hp=99 WHERE stats.id_character=id_char;
    END IF;

END LOOP bucle;
CLOSE cur;
```

2. HANDLERS

En SQL, los handlers (manejadores) son una característica que se utiliza para manejar excepciones y errores durante la ejecución de una consulta o un bloque de código. Los handlers permiten definir acciones personalizadas que se ejecutarán en respuesta a un tipo específico de error o excepción.

Cuando se produce un error o una excepción en SQL, el sistema genera un mensaje de error y puede detener la ejecución del código. Sin embargo, mediante el uso de handlers, se puede interceptar y controlar estos errores, evitando que se interrumpa el flujo normal del programa.

Acciones que se pueden realizar dentro de un handler:

- Imprimir mensajes de error personalizados.
- Realizar operaciones de recuperación de datos.
- Ejecutar consultas o procedimientos almacenados.
- Continuar con la ejecución normal del programa a pesar del error.

```
DELIMITER $$

CREATE PROCEDURE conspiraciones (_id_conspiration INT, _id_user INT, id_telacreiste INT)
BEGIN

    DECLARE handler_p INT DEFAULT 1;
    DECLARE EXIT HANDLER FOR SQLEXCEPTION SET handler_p = 0;

    START TRANSACTION;
        INSERT INTO users_conspiraciones (id_user, id_conspiration)
        VALUES (_id_user, _id_conspiration)

        IF id_telacreiste = 1 THEN
            COMMIT;
        ELSE
            ROLLBACK;
        END IF;

    END $$

DELIMITER ;
```

3. VIEWS

Una vista en SQL es una tabla virtual que se genera a través de una o más tablas existentes en la base de datos. Esta no contiene datos físicos, pero actúa como una tabla lógica que se puede consultar y manipular como cualquier otra tabla. Al crear una vista, se define una consulta que especifica los campos y condiciones de filtrado.

```
DROP VIEW IF EXISTS sum_cost_medicines;  
  
CREATE VIEW sum_cost_medicines AS  
SELECT SUM(total_coste) AS sumatorio_total_coste  
FROM coste_total_medicinas;
```

4. USUARIOS

Cuando se crea un usuario en un sistema y se le otorgan permisos, no se le proporcionan todos los permisos disponibles. Algunos permisos están reservados exclusivamente para el usuario principal del sistema, conocido como "root" o superusuario. Esto se hace para evitar posibles errores o problemas al otorgar permisos a usuarios que no son parte de la administración del sistema.

Los permisos exclusivos del usuario root solo pueden ser otorgados por él mismo. Si un usuario intenta otorgar permisos a otro usuario, solo puede dar los permisos que él mismo posee. Por ejemplo, si un usuario tiene permisos para editar, leer y otorgar permisos, solo podrá otorgar permisos para leer y editar a otros usuarios. No tendrá la capacidad de otorgar los permisos de otorgar permisos a otros.