

## 2. Sentencias DML

### 2.1. Manipulación de datos con SQL

SQL corresponde al acrónimo de *Structured Query Language* (Lenguaje Estructurado de Consultas). Aunque en un principio fue creado para hacer consultas, se utiliza para controlar todas las funciones que suministra un SGBD a sus usuarios, incluyendo todas las funciones propias de los lenguajes diseñados para el manejo de bases de datos: Lenguaje de Definición de Datos (DDL), Lenguaje de Manipulación de Datos (DML) y Lenguaje de Control de Datos (DCL).

El DML permite realizar las operaciones necesarias para manejar los datos almacenados en una base de datos. Estas operaciones consisten en insertar filas de datos (INSERT), modificar el contenido de las filas de datos (UPDATE) o borrar filas de datos (DELETE).

### 2.2. Sentencia INSERT

La sentencia INSERT permite insertar nuevas filas en tablas existentes. Los valores a insertar se pueden escribir como un conjunto de valores entre paréntesis para cada fila nueva. Por ejemplo:

```
insert into nombre_tabla (col1, col2, col3)
values (15, null, default),
       (25, 'Prueba1', null),
       (55, 'Prueba2', 'Prueba3');
```

La sentencia anterior inserta tres filas nuevas en la tabla. Los valores que se ponen entre paréntesis se corresponden con las columnas especificadas antes de la cláusula VALUES. A modo gráfico, tras ejecutar la sentencia anterior la tabla quedaría tal que así:

col1	col2	col3
15	null	default
25	'Prueba1'	null
55	'Prueba2'	'Prueba3'

En el caso de que estemos insertando valores para todas las columnas de la tabla, no necesitamos especificar el nombre de las columnas. Por ejemplo:

```
insert into nombre_tabla
values (15, null, default),
       (25, 'Prueba1', null),
       (55, 'Prueba2', 'Prueba3');
```

Podemos consultar la documentación de la sentencia INSERT en:

[dev.mysql.com/doc/refman/8.0/en/insert.html](https://dev.mysql.com/doc/refman/8.0/en/insert.html)

## 2.3. Sentencia UPDATE

Permite modificar los datos contenidos en las tablas. La sintaxis es:

```
UPDATE nombre_tabla
SET nombre_columna = {expresión | DEFAULT | NULL}
    [, nombre_columna = {expresión | DEFAULT | NULL}] [, ...]
[WHERE condición]
[ORDER BY expresión]
[LIMIT número_filas]
```

- La cláusula SET indica las columnas que se van a modificar y los valores que van a recibir.
- La cláusula WHERE es opcional y especifica las filas que deben actualizarse. Si no se escribe la cláusula WHERE, se actualizarán todas las filas de la tabla.
- La cláusula ORDER BY es opcional e indica el orden en el que se actualizan las filas.
- La cláusula LIMIT es opcional y establece el número de filas a actualizar.

Un ejemplo de sentencia UPDATE sería:

```
update nombre_tabla
set col2 = 'texto1', col3 = null
where col1 > 30;
```

Esta sentencia modificaría los valores de las columnas *col2* y *col3* para aquellas filas cuyo valor de la columna *col1* sea mayor que 30.

Podemos consultar la documentación de la sentencia UPDATE en:

[dev.mysql.com/doc/refman/8.0/en/update.html](https://dev.mysql.com/doc/refman/8.0/en/update.html)

## 2.4. Sentencia DELETE

Permite eliminar filas de las tablas. La sintaxis es:

```
DELETE FROM nombre_tabla
[WHERE condición]
[ORDER BY expresión]
[LIMIT número_filas]
```

Las cláusulas WHERE, ORDER BY y LIMIT tienen la misma función que en la sentencia UPDATE.

Un ejemplo de sentencia DELETE:

```
delete from nombre_tabla
where coll > 30;
```

Esta sentencia borraría aquellas filas que tengan en la columna *coll* un valor mayor que 30.

Podemos consultar la documentación de la sentencia DELETE en:

[dev.mysql.com/doc/refman/8.0/en/delete.html](http://dev.mysql.com/doc/refman/8.0/en/delete.html)

## 2.5. Uso de subconsultas en las sentencias DML

Es posible utilizar subconsultas en la cláusula WHERE de una sentencia INSERT, UPDATE o DELETE.

Por ejemplo, con una sentencia INSERT, queremos añadir en una tabla de aprobados a aquellos alumnos que tengan una nota igual o superior a 5:

```
insert into aprobados(dni, nota)
select dni, nota from alumnos where nota >= 5;
```

Con una sentencia UPDATE, queremos modificar el salario de los empleados que trabajen en un departamento de la provincia de Pontevedra:

```
update empleado
set salario = 1000
where departamento in (select codigo
                        from departamento
                        where provincia = "Pontevedra");
```

Con una sentencia DELETE, queremos eliminar a aquellos empleados que cobren el salario más alto de la empresa:

```
delete from empleado
where salario = (select max(salario) from empleado);
```