



Bases de datos - Relacionales

► Sentencias SQL

// Conceptos de lenguaje estructurado de consultas de base de datos

¿Qué es SQL?

SQL (Structured Query Language - Lenguaje Estructurado de Consulta): es un lenguaje estándar utilizado para crear, modificar, mantener y consultar una base de datos relacional. Es un estándar ANSI, esto implica que conociendo este lenguaje se puede manipular cualquier base de datos de cualquier fabricante (claramente, cada fabricante tiene sus particularidades y extensiones, pero la sintaxis es similar en todos los casos).

DDL y DML

SQL se puede dividir en dos lenguajes DDL y DML. Cada parte tiene sentencias/comandos para realizar operaciones de distinta naturaleza. Estos dos grupos son:

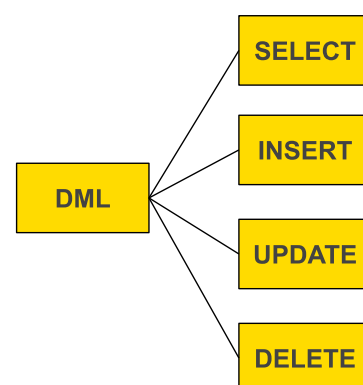
DDL (Data Definition Language): Nos permitirán crear nuevas bases de datos, modificarlas y eliminarlas. Estos comandos afectan a tablas, campos e índices.

- CREATE Crea nuevas tablas, campos e índices.
- ALTER Modifica las tablas
- DROP Elimina tablas e índices.

DML (Data Manipulation Language): Nos permite manipular, consultar, ordenar, filtrar, etc. los datos existentes en una base de datos.

- SELECT Consulta registros en la base de datos
- INSERT Inserta un nuevo registro
- UPDATE Modifica valores de un registro
- DELETE Elimina registros

Se desarrollará de forma más específica, las sentencias SQL DML.



SELECT

Se utiliza esta sentencia para recuperar/mostrar datos de una tabla.

El formato general al usar este comando es:

SELECT <columnas separadas por comas> **FROM** <tabla>;

Teniendo en cuenta que se tiene una tabla *Cliente*:

Cliente (id_cliente, dni, nombre, apellido, domicilio);

Ejemplos:

SELECT * FROM Cliente;

//Recupera todas las columnas de la tabla; sin establecer cláusula WHERE, se obtienen todos los registros de la tabla Cliente.

SELECT nombre **FROM** Cliente;

//Recupera todos los nombres de la tabla.

SELECT apellido, domicilio **FROM** Cliente;

//Recupera todos los apellidos con su domicilio de la tabla.

WHERE: Se utiliza esta sentencia para proponer la condición específica que debe cumplir el dato que se pretende traer.

Ejemplo:

SELECT apellido, domicilio **FROM** Cliente **WHERE** nombre='Paolo';

//De esta forma, estaría mostrando el apellido y domicilio de todos los clientes que tengan el nombre Paolo.

INSERT

Se utiliza esta sentencia para incorporar datos a una tabla.

El formato general al usar este comando es:

INSERT INTO <tabla> **VALUES** (dato 1, dato 2, dato n);

Ejemplos:

Teniendo en cuenta que la tabla *Cliente* tiene la siguiente estructura:

Cliente (id_cliente INTEGER, dni VARCHAR, nombre VARCHAR, apellido VARCHAR, domicilio VARCHAR);

Para insertar filas en la tabla:

INSERT INTO Cliente **VALUES** (0001, '00111000', 'Andriy', 'Shevchenko', 'Ucrania 1613');

INSERT INTO Cliente **VALUES** (0002, '00999111', 'Andrea', 'Pirlo', 'Italia 2006');

INSERT INTO Cliente **VALUES** (0003, '00111222', 'Paolo', 'Maldini', 'Milan 2007');

UPDATE

Se utiliza esta sentencia para modificar datos en una tabla.

El formato general al usar este comando es:

UPDATE <tabla> **SET** datoaModificar='NuevoValor' **WHERE** dato='Valor';

Ejemplo:

UPDATE Cliente **SET** domicilio='Turín 2012' **WHERE** dni='00999111';

//Continuando con el ejemplo citado en el apartado anterior, se estaría modificando el domicilio de Andrea Pirlo.



Si no colocamos la cláusula WHERE, se modificarán todos los domicilios de la tabla Cliente.

DELETE

Se utiliza esta sentencia para eliminar registros en una tabla.

El formato general al usar este comando es:

DELETE FROM <tabla> **WHERE** dato='Valor';

Ejemplo:

DELETE FROM Cliente **WHERE** id_cliente=0001;

//Continuando con el ejemplo citado en los apartados anteriores, se estaría eliminando el registro completo del cliente Andriy Shevchenko.



Si no colocamos la cláusula WHERE, se eliminarán todos los registros de la tabla Cliente.