

GESTIÓN DE BASE DE DATOS

UNIDAD 5: SQL



NOELIA HUGUET CHACÓN

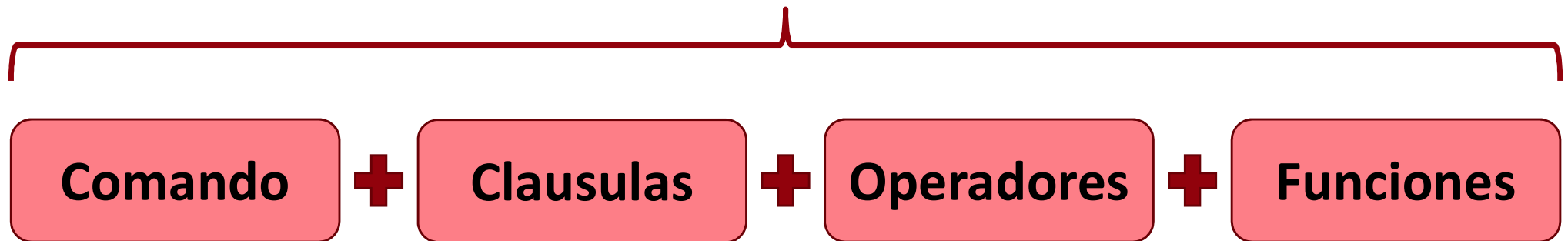


DML (Data Manipulation Language)

COMANDOS DE DML (Date Manipulation Language)	
COMANDO	DESCRIPCIÓN
<i>SELECT</i>	Encargado de consultar registros de la base de datos que satisfagan un criterio determinado.
<i>INSERT</i>	Encargado de cargar lotes de datos en la base de datos en una única operación.
<i>UPDATE</i>	Encargado de modificar los valores de los campos y registros especificados.
<i>DELETE</i>	Encargado de eliminar los registros de una tabla.



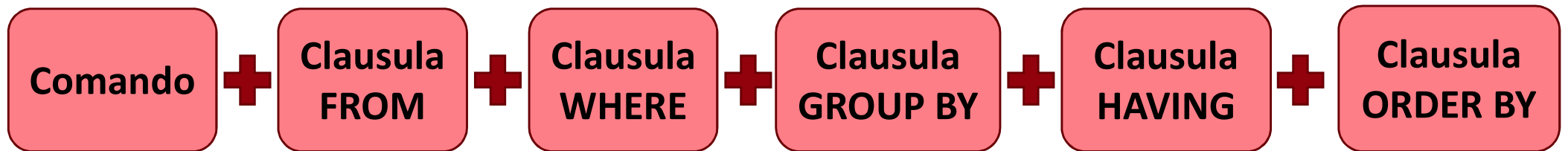
INSTRUCCIÓN SQL



No es necesario todo, solo Comando + Clausula



DML (Data Manipulation Language)





SELECT

- **Se usa para:**
 - **Obtener datos de ciertas columnas de una tabla.**
 - **Obtener registros de una tabla siguiendo unos criterios definidos.**
 - **Mezclar datos de distintas tablas.**



SELECT

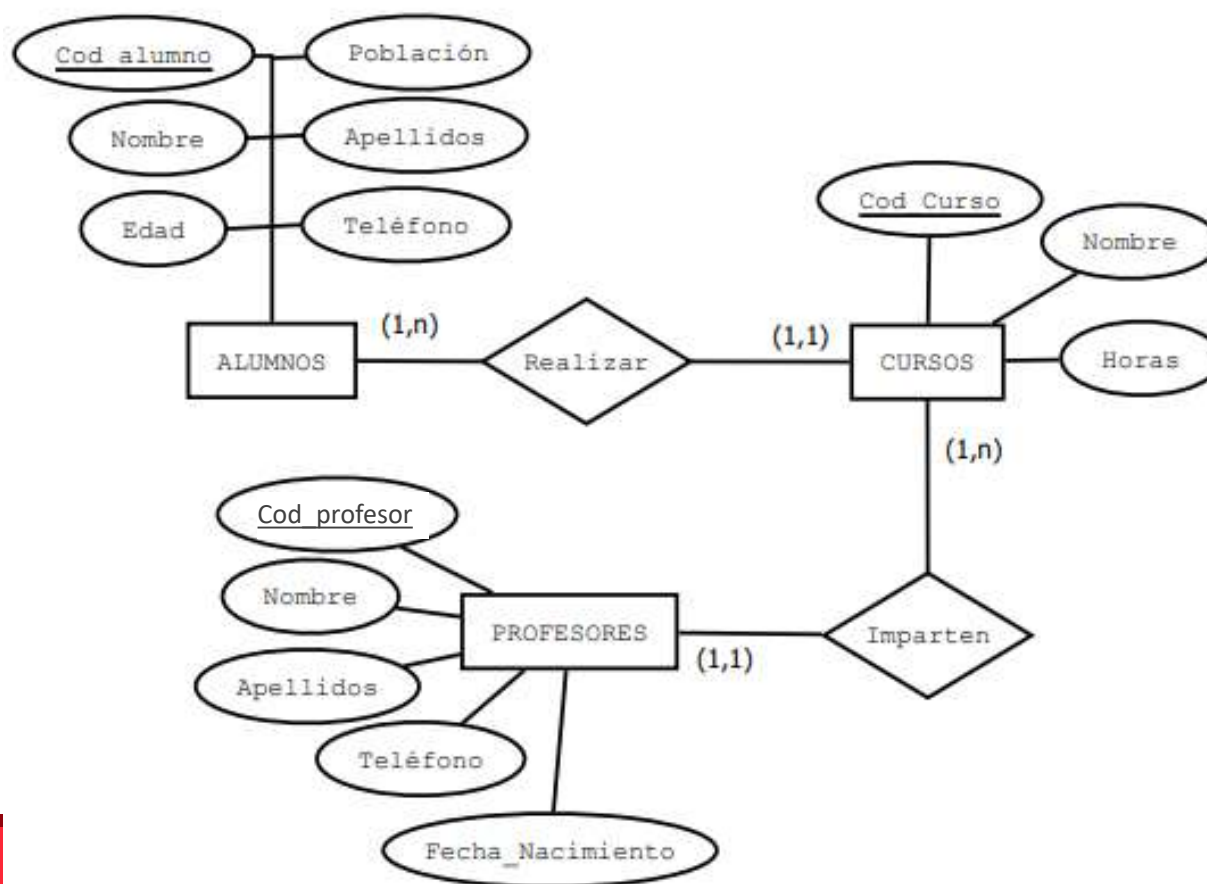
SELECT * | {[DISTINCT] columna | expresión [[AS] alias], ...} FROM tabla;

- * → Se seleccionan todas las columnas
- DISTINCT → No se muestran valores duplicados
- Columna → nombre de la columna que se quiere mostrar
- Expresión → expresión válida en SQL
- Alias → nombre que se le puede dar a una columna



SELECT ... FROM

■ EJEMPLOS:





SELECT ... FROM

■ EJEMPLOS:

SELECT Nombre, Apellidos, Edad **FROM** Alumnos;

SELECT * **FROM** Alumnos;

SELECT DISTINCT Población **FROM** Alumnos;

SELECT EDAD **AS** AÑOS **FROM** Alumnos



SELECT ... WHERE

- **WHERE:** es la clausula que se usa para establecer un criterio o varios de filtrado de la información.

SELECT campos **FROM** tabla **WHERE** condición;

- Se utilizan los operadores para hacer comparaciones de números, textos, fechas...
- En el caso de textos, se establece el orden alfabético estricto, es decir: Mayúsculas antes que minúsculas y la ñ y vocales acentuadas no están ordenadas.



SELECT ... WHERE

■ EJEMPLOS:

SELECT Nombre, Apellidos, Edad **FROM** Alumnos **WHERE** Edad>18;

SELECT Nombre, Apellidos, Edad **FROM** Alumnos **WHERE** Edad>=12 **AND** Edad<=17;

SELECT Nombre, Apellidos, Edad **FROM** Alumnos **WHERE** Edad>25 **OR** Edad<18;



SELECT ... WHERE

■ EJEMPLOS:

SELECT Nombre, Apellidos **FROM** Profesores **WHERE** Nombre = 'Carlos';

SELECT Nombre, Apellidos **FROM** Alumnos **WHERE** Nombre = 'Daniel' **AND** Edad < 20;



SELECT ... WHERE

■ EJEMPLOS:

SELECT Nombre, Fecha_Nacimiento **FROM** Profesores **WHERE** Fecha_Nacimiento < '1980/01/01';

SELECT Nombre, Fecha_Nacimiento **FROM** Profesores **WHERE** Fecha_Nacimiento **BETWEEN** '1980/01/01' **AND** '1990/01/01';

SELECT Nombre, Fecha_Nacimiento **FROM** Profesores **WHERE** Fecha_Nacimiento >= '1980/01/01' **AND** Fecha_Nacimiento <= '1990/01/01';



SELECT ... ORDER BY

- Se utiliza para que los registros seleccionados se muestren ordenados ascendentemente o descendentemente según el campo/s elegido/s.

SELECT campos **FROM** tabla **WHERE** condición
ORDER BY campos para ordenar **ASC/DESC**;



SELECT ... ORDER BY

- **Ejemplo 1:** Obtener los alumnos ordenados por apellidos y luego por el nombre.

```
SELECT * FROM Alumnos ORDER BY Apellidos, Nombre;
```

- **Ejemplo 2:** Obtener los alumnos ordenados por apellidos y luego por el nombre de forma descendente.

```
SELECT * FROM Alumnos ORDER BY Apellidos DESC , Nombre DESC ;
```