

DDL y DML

Las sentencias SQL se dividen en dos categorías; **Lenguaje de definición de datos**; data definition language (**DDL**) y **Lenguaje de manipulación de datos**; data manipulation language (**DML**).

Lenguaje de definición de datos (DDL)

Las sentencias DDL se utilizan para crear y modificar la estructura de las tablas así como otros objetos de la base de datos.

- CREATE - para crear objetos en la base de datos.
- ALTER - modifica la estructura de la base de datos.
- DROP - borra objetos de la base de datos.
- TRUNCATE - elimina todos los registros de la tabla, incluyendo todos los espacios asignados a los registros.

Lenguaje de manipulación de datos (DML)

Las sentencias de lenguaje de manipulación de datos (DML) son utilizadas para gestionar datos dentro de los schemas. Algunos ejemplos:

- SELECT - para obtener datos de una base de datos.
- INSERT - para insertar datos a una tabla.
- UPDATE - para modificar datos existentes dentro de una tabla.
- DELETE - elimina todos los registros de la tabla; no borra los espacios asignados a los registros.

Sentencia SELECT

La sentencia SELECT se utiliza para seleccionar datos de una base de datos.

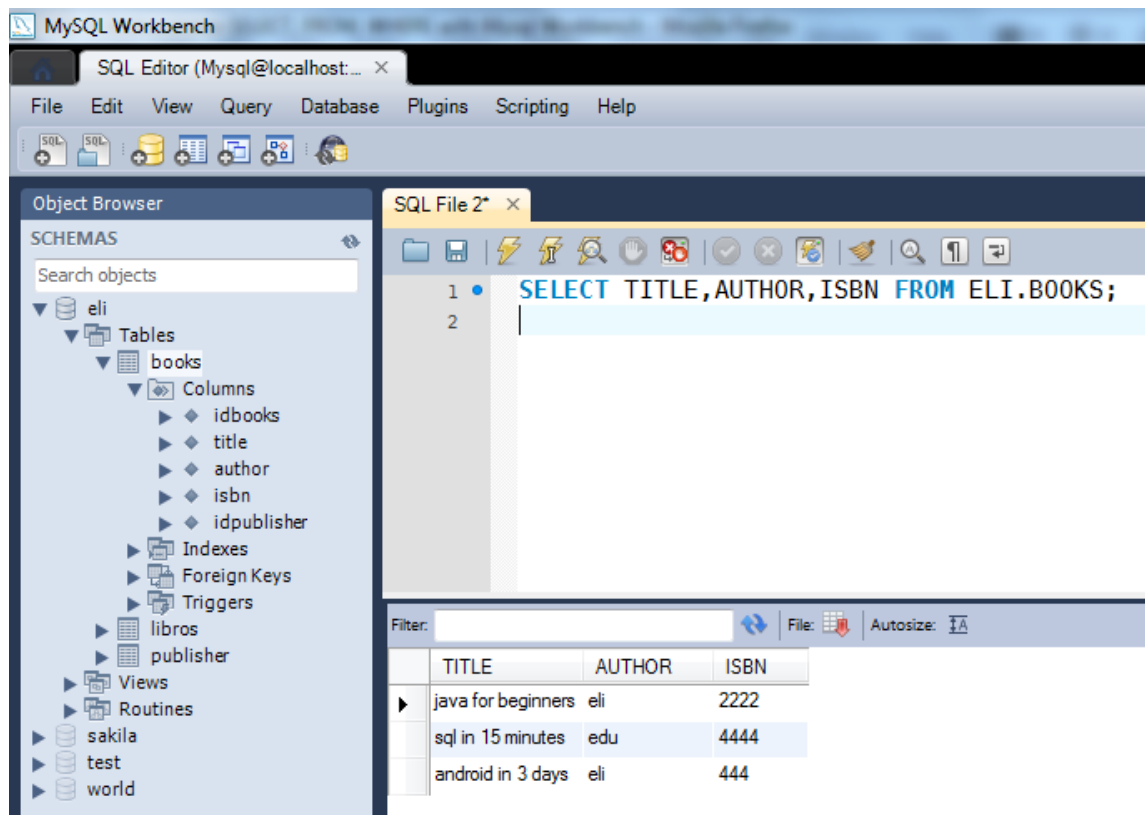
Se guarda el resultado en una tabla llamada "result-set".

Sintaxis de la Sentencia SELECT 1

```
SELECT column_name,column_name  
FROM table_name;
```

Ejemplo de la sentencia SELECT

En el siguiente ejemplo pedimos el "titulo", "autor" y "isbn" de todos los registros de la tabla "books". Como ven SQL no hace distinción entre mayúsculas y minúsculas. Podemos escribir SQL tanto en minúsculas como mayúsculas.



y

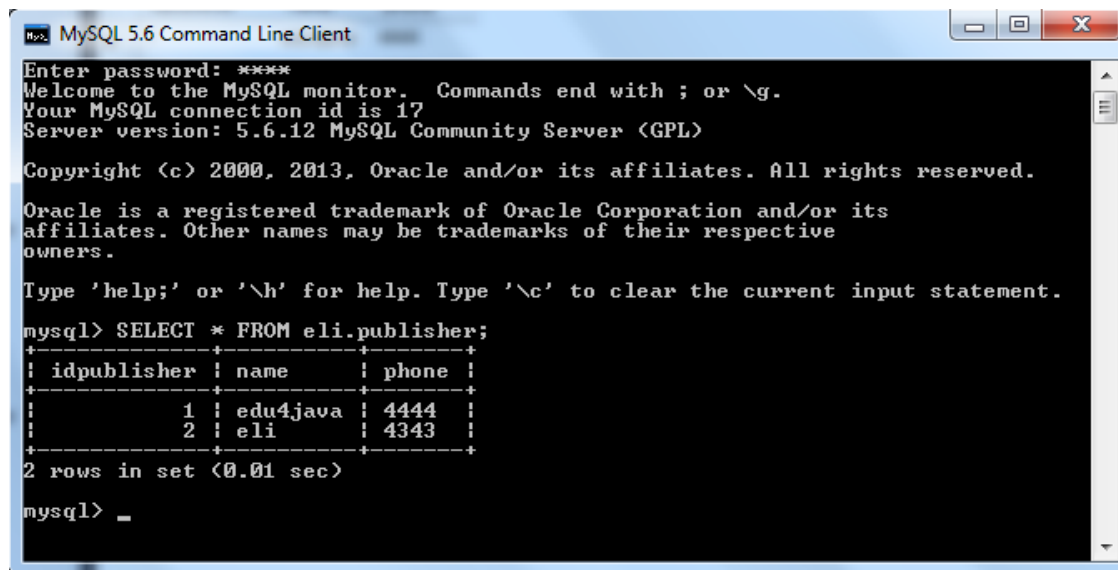
Sintaxis de la Sentencia SELECT 2

```
SELECT * FROM table_name;
```

EL asterisco * significa que queremos todas las columnas de la tabla.

Ejemplo de la sentencia SELECT * FROM

A continuación podemos ver el resultado de la petición; `SELECT * FROM eli.publisher;` toda la información de la tabla "publisher";



Como ven, estamos utilizando indistintamente la consola y el workbench. Es importante que la consola les sea familiar, ya que así la pueden utilizar cuando no tengan una interface gráfica.

Sentencia SQL WHERE

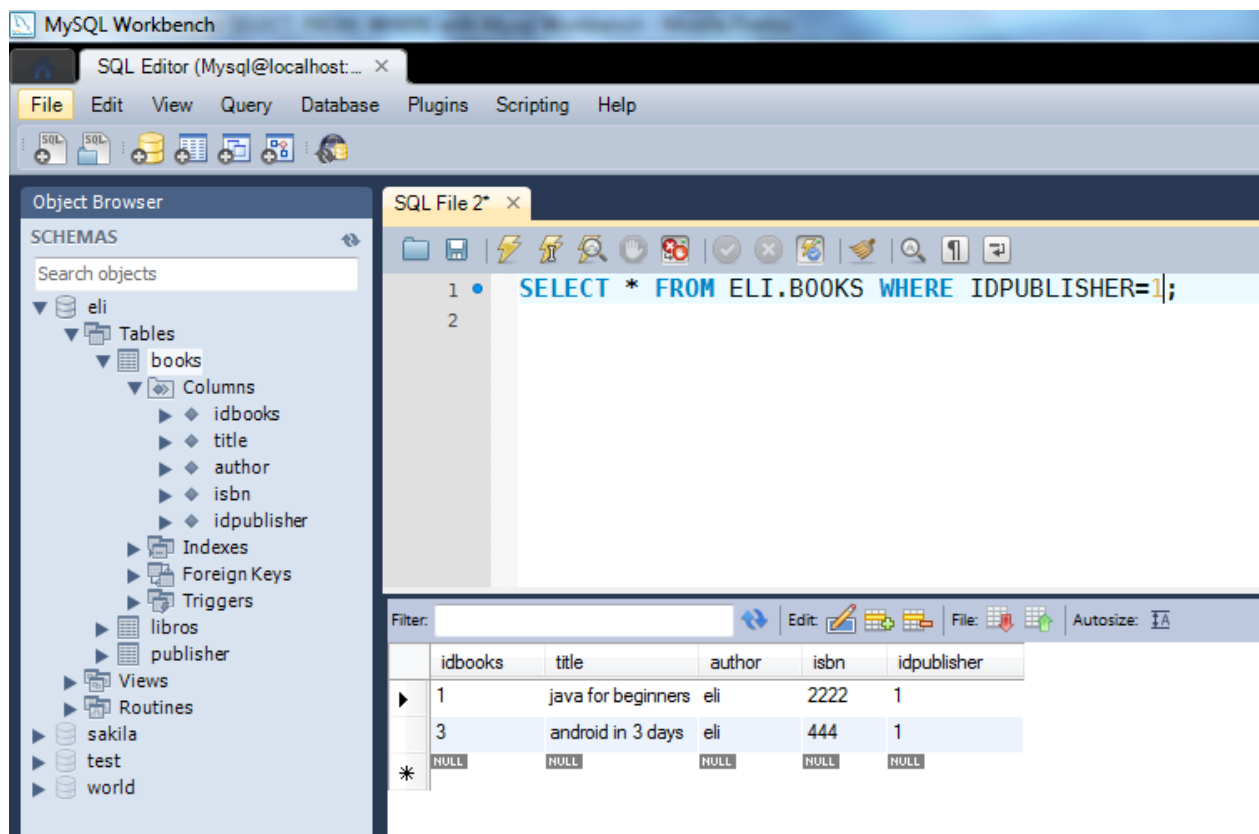
La sentencia WHERE se usa para extraer sólo los registros que cumplan con una condición. Funciona como un filtro.

Sintaxis de la sentencia SQL WHERE

```
SELECT column_name,column_name
FROM table_name
WHERE column_name operator value;
```

Ejemplo de la sentencia SQL WHERE

En este ejemplo, estamos pidiendo todos los registros de la tabla "books" que tengan "idpublisher"=1. Conseguiremos los resultados que cumplan esa condición;



Claúsula ORDER BY

La cláusula ORDER BY se utiliza para ordenar los resultados a través de una o más columnas.

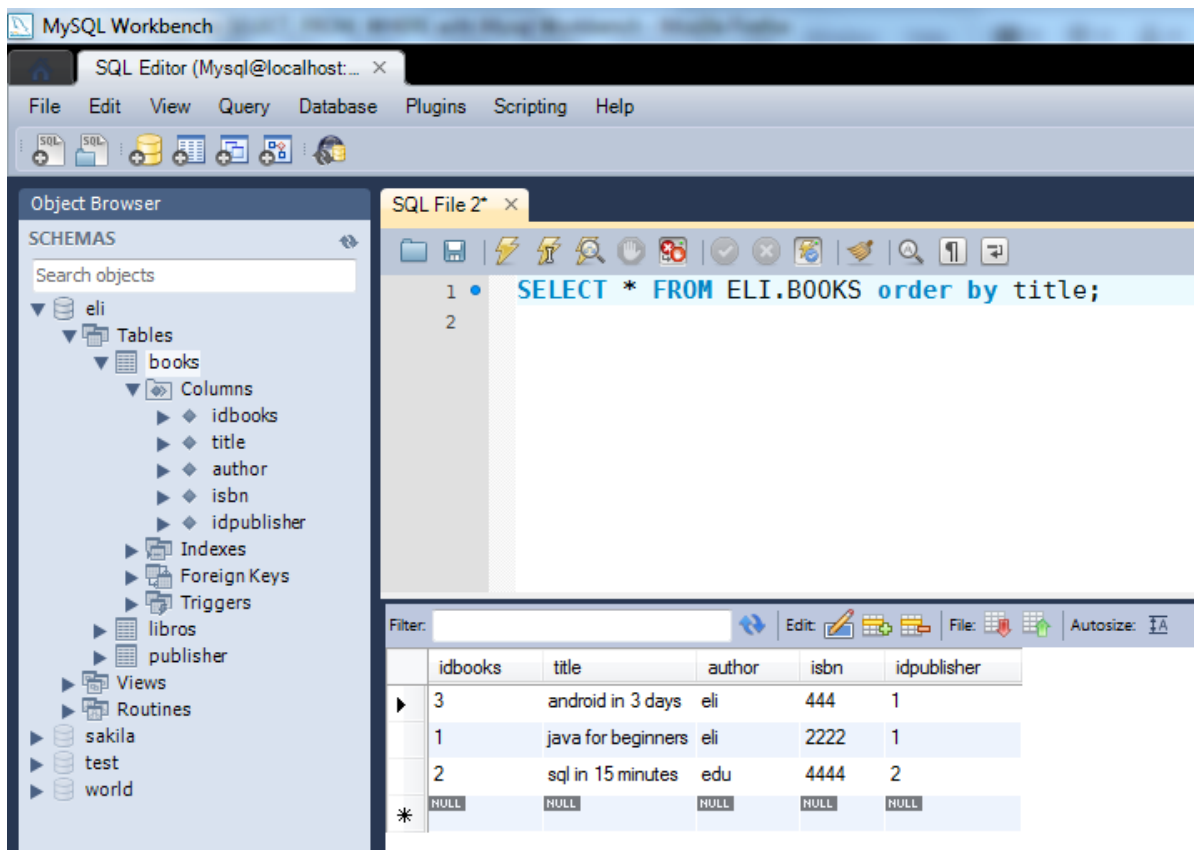
La cláusula ORDER BY ordena los registros de manera ascendente por defecto. Para hacerlo de manera descendente, se puede utilizar la cláusula DESC.

Sintaxis de la cláusula SQL ORDER BY

```
SELECT column_name,column_name
FROM table_name
ORDER BY column_name,column_name ASC|DESC;
```

Ejemplo de la cláusula ORDER BY

En este ejemplo ordenamos la tabla "books" en orden alfabético utilizando el "título";



The screenshot shows the MySQL Workbench interface. On the left, the Object Browser displays the database structure, including the 'eli' database with a 'books' table. The 'books' table has columns: idbooks, title, author, isbn, and idpublisher. The main SQL Editor window shows the query: `SELECT * FROM ELI.BOOKS order by title;`. Below the editor, the results are displayed in a table format.

	idbooks	title	author	isbn	idpublisher
▶	3	android in 3 days	eli	444	1
	1	java for beginners	eli	2222	1
	2	sql in 15 minutes	edu	4444	2
*	NULL	NULL	NULL	NULL	NULL

[<< Anterior](#) [Siguiente >>](#)