
BASE DE DATOS

Continuamos con nuestra
aventura digital !!



Casa del futuro

En este taller conocerán uno de los motores más populares y gratuitos de base de datos y aprenderán a crear, diseñar y modelar Bases de Datos usando los diagramas de Entidad Relación (DER) con MySQL Workbench.

DATE: 23/06/2023







ÍNDICE

01

SELECT

02

**WHERE
ORDER BY**

03

**BETWEEN
LIKE**



01

SELECT

Cómo usarlo

Toda consulta a la base de datos va a empezar con la palabra **SELECT**.

Su funcionalidad es la de realizar consultas sobre **una** o **varias columnas** de una tabla.

Para especificar sobre qué tabla queremos realizar esa consulta usamos la palabra **FROM** seguida del nombre de la tabla.

SQL

```
SELECT nombre_columna, nombre_columna, ...  
FROM nombre_tabla;
```

Ejemplo - Tabla Peliculas

id	titulo	rating	fecha_estreno	pais
1001	Pulp Fiction	9.8	1995-02-16	Estados Unidos
1002	Kill Bill	9.5	2003-11-27	Estados Unidos

De esta tabla completa, para conocer solamente los títulos y ratings de las películas guardadas en la tabla **películas**, podríamos hacerlo ejecutando la siguiente consulta:

SQL

```
SELECT id, titulo, rating  
FROM peliculas;
```



02

WHERE ORDER BY

WHERE

La funcionalidad del **WHERE** es la de condicionar y filtrar las consultas **SELECT** que se realizan a una base de datos.

SQL

```
SELECT nombre_columna_1, nombre_columna_2, ...  
FROM nombre_tabla  
WHERE condicion;
```

Teniendo una tabla **clientes**, podría consultar primer nombre y apellido, filtrando con un **WHERE** solamente los usuarios **que su país es igual a Argentina** de la siguiente manera:

SQL

```
SELECT primer_nombre, apellido  
FROM clientes  
WHERE pais = 'Argentina';
```

Operadores

=► Igual a
>► Mayor que
>=► Mayor o igual que
<► Menor que
<=► Menor o igual que
<>► Diferente a
!=► Diferente a

IS NULL► Es nulo
BETWEEN► Entre dos valores
IN► Lista de valores
LIKE► Se ajusta a...

Queries de ejemplo

SQL

```
SELECT primer_nombre, apellido  
FROM clientes  
WHERE pais <> 'Argentina';
```

SQL

```
SELECT primer_nombre, apellido  
FROM clientes  
WHERE id < 15;
```

SQL

```
SELECT primer_nombre, apellido  
FROM clientes  
WHERE id > 5;
```

SQL

```
SELECT *  
FROM canciones  
  WHERE id >= 3  
  AND id < 8;
```

SQL

```
SELECT *  
FROM canciones  
  WHERE id = 2  
  OR id = 6;
```

SQL

```
DELETE FROM usuarios  
WHERE id = 2;
```



Si en esta query quitáramos el WHERE...
!!!Borraríamos toda la tabla!!!

ORDER BY

ORDER BY se utiliza para ordenar los resultados de una consulta **según el valor de la columna especificada**. Por defecto, se ordena de forma ascendente (ASC) según los valores de la columna. También se puede ordenar de manera descendente (DESC) aclarándolo en la consulta.

SQL

```
SELECT nombre_columna1, nombre_columna2  
FROM tabla  
WHERE condicion  
ORDER BY nombre_columna1;
```

Query de ejemplo

Teniendo una tabla **usuarios**, podría consultar los nombres, filtrar con un **WHERE** solamente los usuarios **mayores de 21 años** y ordenarlos de forma descendente tomando como referencia la columna nombre.

SQL

```
SELECT nombre, rating
FROM artistas
WHERE rating > 1.0
ORDER BY nombre DESC;
```



03

**BETWEEN
LIKE**

BETWEEN

Cuando necesitamos obtener valores **dentro de un rango**, usamos el operador BETWEEN.

- BETWEEN **incluye** los **extremos**.
- BETWEEN funciona con **números, textos y fechas**.
- Se usa como un filtro de un WHERE.

Por ejemplo, coloquialmente:

- Dados los números: 4, 7, 2, 9, 1

Si hiciéramos un BETWEEN entre 2 y 7 devolvería 4, 7, 2 (*excluye el 9 y el 1, e incluye el 2*).

Query de ejemplo

Con la siguiente consulta estaríamos seleccionando **nombre** y **edad** de la tabla **alumnos** sólo cuando las edades estén **entre** 6 y 12.

SQL

```
SELECT nombre, edad  
FROM alumnos  
WHERE edad BETWEEN 6 AND 12;
```

LIKE

Cuando hacemos un filtro con un **WHERE**, podemos especificar un patrón de búsqueda que nos permita especificar algo concreto que queremos encontrar en los registros. Eso lo logramos utilizando **comodines** (*wildcards*).

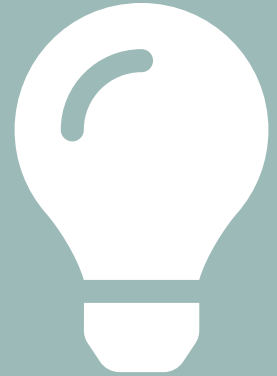
Por ejemplo, podríamos querer buscar:

- Los nombres que tengan la letra 'a' como segundo carácter.
 - Las direcciones postales que incluyan la calle 'Monroe'.
 - Los clientes que empiecen con 'Los' y terminen con 's'.
-

“

COMODÍN %

Es un sustituto que representa
cero, uno, o varios caracteres.



”

“

COMODÍN _

Es un sustituto para **un solo**
carácter.



”

Queries de ejemplo

SQL

```
SELECT nombre  
FROM usuarios  
WHERE nombre LIKE '_a%';
```

Devuelve aquellos nombres que tengan la letra 'a' como segundo carácter.

SQL

```
SELECT nombre  
FROM usuarios  
WHERE direccion LIKE '%Monroe%';
```

Devuelve las direcciones de los usuarios que incluyan la calle 'Monroe'.

Queries de ejemplo

SQL

```
SELECT nombre  
FROM clientes  
WHERE nombre LIKE 'Los%s';
```

Devuelve los clientes que empiecen con 'Los' y terminen con 's'.
