

MySQL

06_ Registro



MySQL_Registro.

Contenido

- | | |
|----------------------------------|----|
| 1. Recuperar registros selectiva | 4 |
| 2. Recuperar registros aleatoria | 8 |
| 3. Reemplazar registros | 10 |

Introducción

Los registros en MySQL se refieren a las filas individuales en una tabla que contienen datos específicos. Cada registro representa una instancia única de los datos almacenados en una tabla, y se identifica mediante una clave primaria única. La clave primaria es un campo o conjunto de campos que identifican de manera única cada registro en la tabla y se utiliza para referenciar y relacionar los datos entre varias tablas.

Cada registro en MySQL se compone de uno o más campos que contienen valores específicos para cada columna en la tabla. Los valores pueden ser de diferentes tipos de datos, como números, cadenas de texto, fechas y horas, entre otros. Los campos también pueden ser opcionales, lo que significa que pueden contener valores nulos si no se proporciona un valor específico para ese campo en el registro.

Los registros en MySQL se pueden insertar, actualizar o eliminar utilizando comandos SQL específicos. Por ejemplo, se puede insertar un nuevo registro en una tabla utilizando el comando INSERT, actualizar un registro existente utilizando el comando UPDATE o eliminar un registro utilizando el comando DELETE.

En resumen, los registros en MySQL son las filas individuales en una tabla que contienen datos específicos. Cada registro se identifica mediante una clave primaria única y se compone de uno o más campos que contienen valores específicos para cada columna en la tabla. Los registros se pueden manipular utilizando comandos SQL específicos.



MySQL_Registro.

1. Cláusula limit del comando select.

La cláusula "limit" se usa para restringir los registros que se retornan en una consulta "select".

Recibe 1 ó 2 argumentos numéricos enteros positivos; el primero indica el número del primer registro a retornar, el segundo, el número máximo de registros a retornar. El número de registro inicial es 0 (no 1).

Si el segundo argumento supera la cantidad de registros de la tabla, se limita hasta el último registro.

Ejemplo:

```
select * from libros limit 0,4;
```

Muestra los primeros 4 registros, 0,1,2 y 3.

Si tipeamos:

```
select * from libros limit 5,4;
```

recuperamos 4 registros, desde el 5 al 8.

Si se coloca un solo argumento, indica el máximo número de registros a retornar, comenzando desde 0. Ejemplo:

```
select * from libros limit 8;
```

Muestra los primeros 8 registros.

Para recuperar los registros desde cierto número hasta el final, se puede colocar un número grande para el segundo argumento:

```
select * from libros limit 6,10000;
```

recupera los registros 7 al último.

"limit" puede combinarse con el comando "delete". Si queremos eliminar 2 registros de la tabla "libros" Usamos:

```
delete from libros  
limit 2;
```

Podemos ordenar los registros por precio (por ejemplo) y borrar 2:

```
delete from libros  
order by precio  
limit 2;
```

esta sentencia borrará los 2 primeros registros, es decir, los de precio más bajo.

Podemos emplear la cláusula "limit" para eliminar registros duplicados. Por ejemplo, continuamos con la tabla "libros" de una librería, ya hemos almacenado el libro "El aleph" de "Borges" de la editorial "Planeta", pero nos equivocamos y volvemos a ingresar el mismo libro, del mismo autor y editorial 2 veces más, es un error que no controla MySQL. Para eliminar el libro duplicado y que sólo quede un registro de él vemos cuántos tenemos:

MySQL_ Registro.

```
select * from libros
where titulo='El aleph' and
autor='Borges' and
editorial='Planeta';
```

Luego eliminamos con "limit" la cantidad sobrante (tenemos 3 y queremos solo 1):

```
delete from libros
where titulo='El aleph' and
autor='Borges' and
editorial='Planeta'
limit 2;
```

Un mensaje nos muestra la cantidad de registros eliminados.

Es decir, con "limit" indicamos la cantidad a eliminar.

Veamos cuántos hay ahora:

```
select * from libros
where titulo='El aleph' and
autor='Borges' and
editorial='Planeta';
```

Sólo queda 1.

Servidor de MySQL instalado en forma local.

Ingreseemos al programa "Workbench" y ejecutemos el siguiente bloque de instrucciones SQL:

```
drop table if exists libros;

create table libros(
codigo int unsigned auto_increment,
titulo varchar(40) not null,
autor varchar(30),
editorial varchar(15),
precio decimal(5,2) unsigned,
primary key (codigo)
);

insert into libros (titulo,autor,editorial,precio)
values('El aleph','Borges','Planeta',15);
insert into libros (titulo,autor,editorial,precio)
values('Martin Fierro','Jose Hernandez','Emece',22.20);
insert into libros (titulo,autor,editorial,precio)
values('Antologia poetica','Borges','Planeta',40);
insert into libros (titulo,autor,editorial,precio)
values('Aprenda PHP','Mario Molina','Emece',18.20);
insert into libros (titulo,autor,editorial,precio)
values('Cervantes y el quijote','Borges','Paidos',36.40);
insert into libros (titulo,autor,editorial,precio)
```

MySQL_ Registro.

```
values('Manual de PHP', 'J.C. Paez', 'Paidos',30.80);
insert into libros (titulo,autor,editorial,precio)
values('Harry Potter y la piedra filosofal','J.K. Rowling','Paidos',45.00);
insert into libros (titulo,autor,editorial,precio)
values('Harry Potter y la camara secreta','J.K. Rowling','Paidos',46.00);
insert into libros (titulo,autor,editorial,precio)
values('Alicia en el pais de las maravillas','Lewis Carroll','Paidos',null);

-- recuperar 4 libros desde el registro cero:
select * from libros limit 0,4;

-- recuperar 4 libros a partir del registro 5
select * from libros limit 5,4;

-- recuperar 8 libros desde el principio
select * from libros limit 8;

-- para recuperar 10000 registros o hasta el final de la tabla a partir
-- del registro 6
select * from libros limit 6,10000;

-- para eliminar los 2 registros con precio más bajo
delete from libros
order by precio
limit 2;

insert into libros (titulo,autor,editorial,precio)
values('El aleph','Borges','Planeta',15);

insert into libros (titulo,autor,editorial,precio)
values('El aleph','Borges','Planeta',15);
insert into libros (titulo,autor,editorial,precio)
values('El aleph','Borges','Planeta',15);

select * from libros
where titulo='El aleph' and
autor='Borges' and
editorial='Planeta';

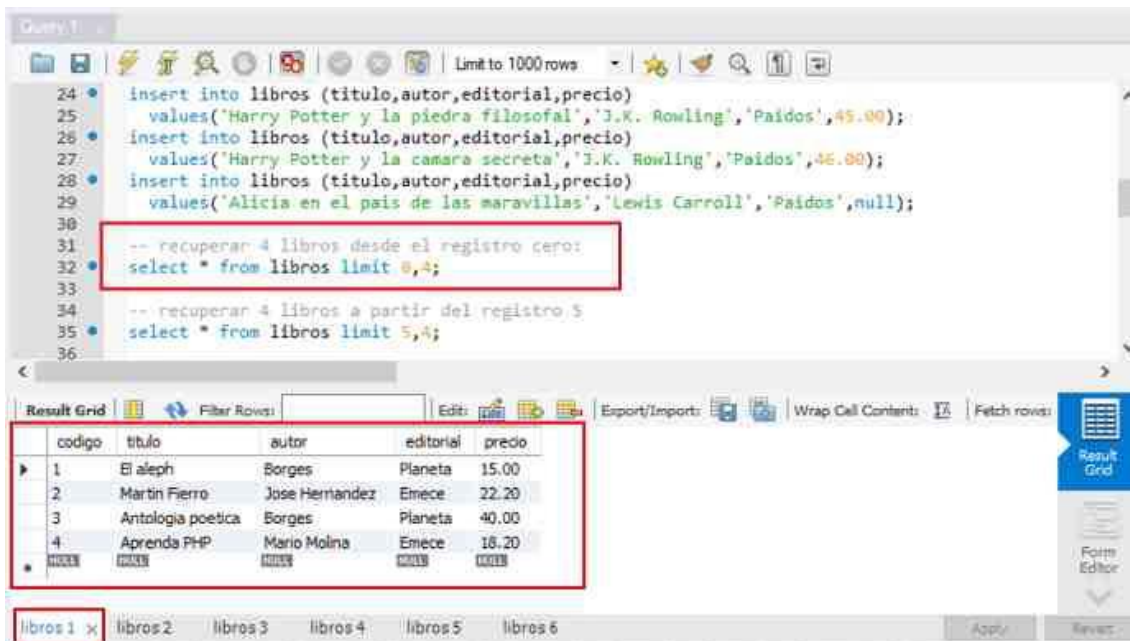
-- eliminamos 2 registros
delete from libros
where titulo='El aleph' and
autor='Borges' and
editorial='Planeta'
limit 2;

select * from libros
```

MySQL_Registro.

where titulo='El aleph' and
autor='Borges' and
editorial='Planeta';

Genera una salida similar a esta:



The screenshot shows a MySQL query editor with the following SQL commands:

```
24 insert into libros (titulo,autor,editorial,precio)
25 values('Harry Potter y la piedra filosofal','J.K. Rowling','Paidos',45.00);
26 insert into libros (titulo,autor,editorial,precio)
27 values('Harry Potter y la camara secreta','J.K. Rowling','Paidos',45.00);
28 insert into libros (titulo,autor,editorial,precio)
29 values('Alicia en el pais de las maravillas','Lewis Carroll','Paidos',null);
30
31 -- recuperar 4 libros desde el registro cero:
32 select * from libros limit 0,4;
33
34 -- recuperar 4 libros a partir del registro 5
35 select * from libros limit 5,4;
36
```

The results of the queries are displayed in a table with the following columns: codigo, titulo, autor, editorial, precio.

codigo	titulo	autor	editorial	precio
1	El aleph	Borges	Planeta	15.00
2	Martin Fierro	Jose Hernandez	Emece	22.20
3	Antologia poetica	Borges	Planeta	40.00
4	Aprenda PHP	Mario Molina	Emece	18.20
NULL	NULL	NULL	NULL	NULL

The table is displayed in a grid view, and the first row is highlighted. The editor also shows a tab labeled 'libros 1' and a 'Fetch rows' button.

MySQL_Registro.

2. Recuperación de registros en forma aleatoria(rand)

Una librería que tiene almacenados los datos de sus libros en una tabla llamada "libros" quiere donar a una institución 5 libros tomados al azar.

Para recuperar de una tabla registros aleatorios se puede utilizar la función "rand()" combinada con "order by" y "limit":

```
select * from libros
order by rand()
limit 5;
```

Nos devuelve 5 registros tomados al azar de la tabla "libros".

Podemos ejecutar la sentencia anterior varias veces seguidas y veremos que los registros recuperados son diferentes en cada ocasión.

Servidor de MySQL instalado en forma local.

Ingresemos al programa "Workbench" y ejecutemos el siguiente bloque de instrucciones SQL:

```
drop table if exists libros;

create table libros(
  codigo int unsigned auto_increment,
  titulo varchar(40),
  autor varchar(30),
  editorial varchar(20),
  precio decimal(5,2) unsigned,
  primary key(codigo)
);

insert into libros values(1,'El aleph','Borges','Planeta',23.5);
insert into libros values(2,'Cervantes y el quijote','Borges','Paidos',33.5);
insert into libros
  values(3,'Alicia a traves del espejo','Lewis Carroll','Planeta',15);
insert into libros
  values(4,'Alicia en el pais de las maravillas','Lewis Carroll','Planeta',18);
insert into libros values(5,'Martin Fierro','Jose Hernandez','Planeta',34.6);
insert into libros values(6,'Martin Fierro','Jose Hernandez','Emece',45);
insert into libros values(7,'Aprenda PHP','Mario Molina','Planeta',55);
insert into libros values(8,'Java en 10 minutos','Mario Molina','Planeta',45);
insert into libros values(9,'Matematica estas ahi','Paenza','Planeta',12.5);

select * from libros
order by rand()
limit 5;

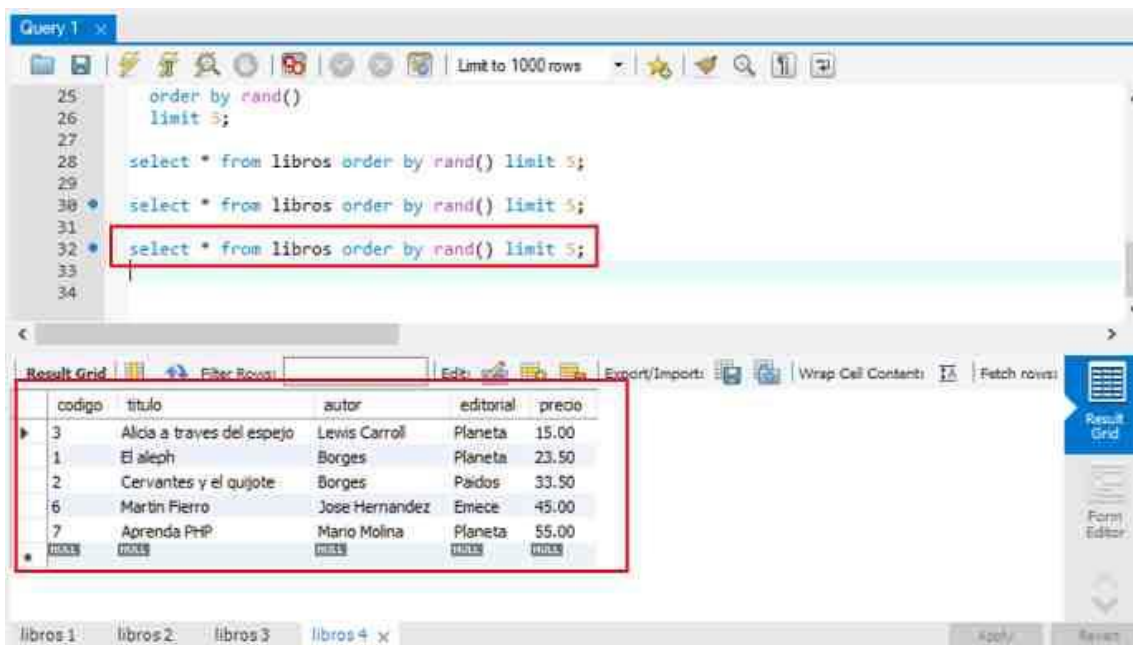
select * from libros order by rand() limit 5;
```


MySQL_Registro.

```
select * from libros order by rand() limit 5;
```

```
select * from libros order by rand() limit 5;
```

Genera una salida similar a esta en una de las llamadas a select:



The screenshot shows a MySQL query editor window titled 'Query 1'. The query editor contains the following SQL code:

```
25: order by rand()  
26: limit 5;  
27:  
28: select * from libros order by rand() limit 5;  
29:  
30: select * from libros order by rand() limit 5;  
31:  
32: select * from libros order by rand() limit 5;  
33:  
34:
```

The query results are displayed in a table with the following columns: codigo, titulo, autor, editorial, and precio. The results are:

codigo	titulo	autor	editorial	precio
3	Alicia a traves del espejo	Lewis Carroll	Planeta	15.00
1	El aleph	Borges	Planeta	23.50
2	Cervantes y el quijote	Borges	Paidós	33.50
6	Martin Fierro	Jose Hernandez	Emecé	45.00
7	Aprenda PHP	Mario Molina	Planeta	55.00

MySQL_Registro.

3. Reemplazar registros (replace)

"replace" reemplaza un registro por otro.

Cuando intentamos ingresar con "insert" un registro que repite el valor de un campo clave o indexado con índice único, aparece un mensaje de error indicando que el valor está duplicado. Si empleamos "replace" en lugar de "insert", el registro existente se borra y se ingresa el nuevo, de esta manera no se duplica el valor único.

Si tenemos la tabla "libros" con el campo "codigo" establecido como clave primaria e intentamos ingresar ("insert") un valor de código existente, aparece un mensaje de error porque no está permitido repetir los valores del campo clave. Si empleamos "replace" en lugar de "insert", la sentencia se ejecuta reemplazando el registro con el valor de código existente por el nuevo registro.

Veamos un ejemplo. Tenemos los siguientes registros almacenados en "libros":

codigo	titulo	autor	editorial	precio
10	Alicia en ..	Lewis Carroll	Emece	15.4
15	Aprenda PHP	Mario Molina	Planeta	45.8
23	El aleph	Borges	Planeta	23.0

Intentamos insertar un registro con valor de clave repetida (código 23):

```
insert into libros values(23,'Java en 10 minutos','Mario Molina','Emece',25.5);
```

Aparece un mensaje de error indicando que hay registros duplicados.

Si empleamos "replace":

```
replace into libros values(23,'Java en 10 minutos','Mario Molina','Emece',25.5);
```

la sentencia se ejecuta y aparece un mensaje indicando que se afectaron 2 filas, esto es porque un registro se eliminó y otro se insertó.

"replace" funciona como "insert" en los siguientes casos:

- si los datos ingresados no afectan al campo único, es decir no se ingresa valor para el campo indexado:

```
replace into libros(titulo,autor,editorial,precio)  
values('Cervantes y el quijote','Borges','Paidos',28);
```

aparece un mensaje indicando que se afectó un solo registro, el ingresado, que se guarda con valor de código 0.

- si el dato para el campo indexado que se ingresa no existe:

```
replace into libros values(30,'Matematica estas ahi','Paenza','Paidos',12.8);
```

aparece un mensaje indicando que se afectó solo una fila, no hubo reemplazo porque el código no existía antes de la nueva inserción.

MySQL_ Registro.

- si la tabla no tiene indexación. Si la tabla "libros" no tuviera establecida ninguna clave primaria (ni índice único), podríamos ingresar varios registros con igual código:

```
replace into libros values(1,'Harry Potter ya la piedra filosofal','Hawking','Emece',48);
```

aparecería un mensaje indicando que se afectó 1 registro (el ingresado), no se reemplazó ninguno y ahora habría 2 libros con código 1.

Servidor de MySQL instalado en forma local.

Ingresemos al programa "Workbench" y ejecutemos el siguiente bloque de instrucciones SQL:

```
drop table if exists libros;

create table libros(
  codigo int unsigned default 0,
  titulo varchar(40),
  autor varchar(30),
  editorial varchar(20),
  precio decimal(5,2) unsigned,
  primary key(codigo)
);

insert into libros
  values (10,'Alicia en el pais de las maravillas','Lewis Carroll','Emece',15.4);
insert into libros
  values (15,'Aprenda PHP','Mario Molina','Planeta',45.8);
insert into libros values (23,'El aleph','Borges','Planeta',23.0);

-- ntentemos ingresar un registro con valor de clave repetida (genera error)
insert into libros values(23,'Java en 10 minutos','Mario Molina','Emece',25.5);

-- Para reemplazar el registro con clave 23 por el nuevo empleamos "replace":
replace into libros values(23,'Java en 10 minutos','Mario Molina','Emece',25.5);

-- ingresamos un registro sin valor de código:
replace into libros(titulo,autor,editorial,precio)
  values('Cervantes y el quijote','Borges','Paidos',28);

-- ingresamos un registro con un valor de código que no existe:
replace into libros values(30,'Matematica estas ahi','Paenza','Paidos',12.8);

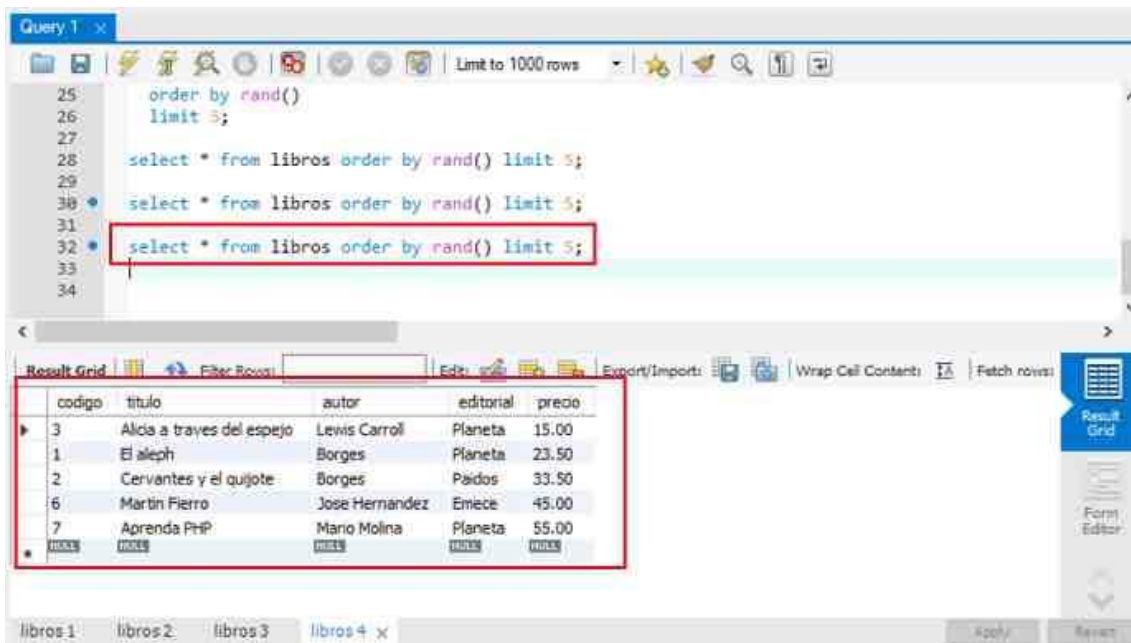
-- Qitemos la clave primaria:
alter table libros drop primary key;

-- Ingresamos un registro con valor de código repetido usando "replace":
replace into libros values(10,'Harry Potter ya la piedra filosofal','Hawking','Emece',48);

select * from libros;
```

MySQL_Registro.

Genera una salida similar a esta:



The screenshot shows a MySQL query editor window. The query editor has a tab labeled "Query 1" and a toolbar with various icons. The query text is as follows:

```
25:     order by rand()
26:     limit 5;
27:
28: select * from libros order by rand() limit 5;
29:
30: select * from libros order by rand() limit 5;
31:
32: select * from libros order by rand() limit 5;
33:
34:
```

The query results are displayed in a table with the following columns: codigo, titulo, autor, editorial, and precio. The results are as follows:

codigo	titulo	autor	editorial	precio
3	Alicia a traves del espejo	Lewis Carroll	Planeta	15.00
1	El aleph	Borges	Planeta	23.50
2	Cervantes y el quijote	Borges	Paidós	33.50
6	Martin Fierro	Jose Hernandez	Emece	45.00
7	Aprenda PHP	Mario Molina	Planeta	55.00
TOTAL	TOTAL	TOTAL	TOTAL	TOTAL

The bottom of the window shows a tab labeled "libros 4" and buttons for "Apply" and "Revert".