

Uso de Join

Qué es un JOIN en MySQL

Los JOIN son usados en una sentencia SQL para recuperar datos de varias tablas al mismo tiempo. Estas tablas tienen que estar relacionadas de alguna forma, por ejemplo, una tabla usuarios, y otra tabla juegos que contiene también la id del usuario al que pertenece el juego:



Tabla Usuarios:

ID	username
----	----------

Tabla Juegos:

ID	juegoname	ID_usuario
----	-----------	------------

Como puedes observar, podríamos asociar cada juego a un usuario mediante su ID. De esta forma con un JOIN uniríamos las dos tablas y extraeríamos en una sola consulta por ejemplo:

-  username
-  juegoname

¿Qué son las tablas relacionales?

Las tablas relacionales, son tablas que se utilizan como intermediarios de otras dos tablas. Normalmente contienen solamente la id de cada uno de los elementos de otras tablas a asociar, y será con lo que trabajaremos para entender los JOINS.

Un ejemplo sacado del de arriba sería tener 3 tablas, usuarios, juegos y juegousuario

Tabla usuario:

ID	username
----	----------

Tabla juegos:

ID	juegoname
----	-----------

Tabla juegousuario:

ID_username	ID_juegoname
-------------	--------------

Ahora piénsalo detenidamente. Si tienes una tabla con todos los juegos que existen, y otra tabla con 200 usuarios, bastará con que cada usuario elija los juegos que tiene. Entonces tú guardas el id de ese usuario y el id del juego que tiene, y un mismo usuario puede tener varios juegos, o varios usuarios tener un juego en concreto.

Lo solucionaríamos con **un doble JOIN** que devolverá los usuarios y juegos por los parámetros que le hayamos pedido.

Nota: Se puede definir un alias para una referencia de tabla por ejemplo: *tbl_name AS alias_name* o *tbl_name alias_name*:

```
SELECT t1.name, t2.salary FROM employee AS t1, info AS t2
WHERE t1.name = t2.name;
```

```
SELECT t1.name, t2.salary FROM employee t1, info t2 WHERE t1.name =
t2.name;
```

¿Qué tipos de join existen?

Tenemos **varios tipos de JOINS**, los vamos a ver uno por uno y cada uno de ellos te servirá para extraer los datos de una forma específica. Dependiendo de lo que necesites tendrás que echar mano de uno u otro, o incluso combinarlos entre ellos, que es lo más complejo.

Ejercicio

Problema:

Una biblioteca registra la información de sus libros en una tabla llamada "libros", los datos de sus socios en "socios" y los préstamos en una tabla "prestamos".

En la tabla "prestamos" se hace referencia al libro y al socio que lo solicita colocando un código que

los identifique.

Crear las siguientes tablas en nuestro Schema prueba:

Creamos las tablas:

```
create table libros(  
  codigo int unsigned auto_increment,  
  titulo varchar(40) not null,  
  autor varchar(20) default 'Desconocido',  
  primary key (codigo)  
);
```

```
create table socios(  
  documento char(8) not null,  
  nombre varchar(30),  
  domicilio varchar(30),  
  primary key (documento)  
);
```

```
create table prestamos(  
  documento char(8) not null,  
  codigolibro int unsigned,  
  fechaprestamo date not null,  
  fechadevolucion date,  
  primary key (codigolibro,fechaprestamo)  
);
```

Ingresamos algunos registros:

```
insert into socios values('22333444','Juan Perez','Colon 345');  
insert into socios values('23333444','Luis Lopez','Caseros 940');  
insert into socios values('25333444','Ana Herrero','Sucre 120');
```

```
insert into libros values(1,'Manual de 2º grado','Molina Manuel');  
insert into libros values(25,'Aprenda PHP','Oscar Mendez');  
insert into libros values(42,'Martin Fierro','Jose Hernandez');
```

```
insert into prestamos values('22333444',1,'2006-08-10','2006-08-12');  
insert into prestamos values('22333444',1,'2006-08-15',null);  
insert into prestamos values('25333444',25,'2006-08-10','2006-08-13');  
insert into prestamos values('25333444',42,'2006-08-10',null);  
insert into prestamos values('25333444',25,'2006-08-15',null);  
insert into prestamos values('30333444',42,'2006-08-02','2006-08-05');  
insert into prestamos values('25333444',2,'2006-08-02','2006-08-05');
```

Al recuperar los datos de los prestamos:

```
select * from prestamos;
```

aparece el código del libro pero no sabemos el nombre y tampoco el nombre del socio sino su documento.

- A) Escribe una query que permita obtener los datos completos de cada préstamo para los cuales encuentra coincidencia en "socios" y "libros", para ello necesitamos consultar las tres tablas, usando un JOIN.
- B) Listar todos los préstamos, incluso los que no encuentran coincidencia en las otras tablas. (Ayuda: usando left join y join)
- C) Listar todos los préstamos con valor coincidente para "libros" pero para "socio" con y sin coincidencia. (ayuda: usando left join)