

75 - Insertar datos en una tabla buscando un valor en otra (insert - select)

Problema:

Un club de dicta clases de distintos deportes a sus socios. Guarda la información de sus socios en una tabla llamada "socios" y las inscripciones en "inscriptos".

1- Elimine las tablas, si existen.

2- Cree las tablas:

```
create table socios(  
  numero int unsigned auto_increment,  
  documento char(8) not null,  
  nombre varchar(30) not null,  
  domicilio varchar(30),  
  primary key(numero)  
);  
  
create table inscriptos(  
  numerosocio int unsigned,  
  deporte varchar(20),  
  año year not null,  
  cuota char(1), /*'s' o 'n', si esta paga o no*/  
  primary key(numerosocio,deporte,año)  
);
```

3- Ingrese los siguientes registros:

```
insert into socios values(23,'22333444','Juan Perez','Colon 123');  
insert into socios values(56,'23333444','Ana Garcia','Sarmiento 984');  
insert into socios values(102,'24333444','Hector Fuentes','Sucre 293');  
insert into socios values(150,'25333444','Agustin Perez','Avellaneda 1234');  
insert into socios values(230,'26333444','Maria Perez','Urquiza 283');  
insert into socios values(231,'29333444','Agustin Perez','Urquiza 283');  
  
insert into inscriptos values(23,'tenis','2015','s');  
insert into inscriptos values(23,'tenis','2016','s');  
insert into inscriptos values(23,'natacion','2015','s');  
insert into inscriptos values(102,'tenis','2015','s');  
insert into inscriptos values(102,'natacion','2016','s');
```

4- El socio con documento "23333444" quiere inscribirse en "basquet" este año, pero no recuerda su número de socio. Inscriba al socio en la tabla "inscriptos" buscando en la tabla "socios" el número de socio a partir del número de documento:

```
insert into inscriptos (numerosocio,deporte,año,cuota)  
select numero,'tenis','2016','s'  
from socios  
where documento='23333444';
```

5- Intente inscribir una persona cuyo número de documento no exista en la tabla "socios":

```
insert into inscriptos (numerosocio,deporte,año,cuota)  
select numero,'tenis','2016','s'  
from socios  
where documento='30333444';
```

No se realiza la inserción porque no encuentra el documento en la tabla "socios".

6- La madre de los socios "Perez" quiere inscribir a su hijo "Agustín" en "basquet" este año, pero no recuerda su número de socio ni su documento. Si se busca el número de socio a partir del nombre ("Agustin Perez"), la consulta retornará 2 registros, porque hay 2 socios llamados "Agustin Perez". Obtendremos como resultado, la inscripción a "basquet" este año, de ambos socios. Si localizamos el

número de socio a partir del domicilio ("Urquiza 283"), también obtendremos 2 inscripciones, porque hay 2 socios con ese domicilio. Ingrese la inscripción del socio, localizando su número de socio con el domicilio:

```
insert into inscriptos (numerosocio,deporte,año,cuota)
select numero,'basquet','2016','n'
from socios
where nombre='Agustin Perez' and
domicilio='Urquiza 283';
```

7- La madre de los socios "Perez" quiere inscribir a ambos hijos "natacion" este año, pero no recuerda sus números de socio ni sus documentos. Inscriba a ambos localizando sus números de socio con el domicilio:

```
insert into inscriptos (numerosocio,deporte,año,cuota)
select numero,'natacion','2016','n'
from socios
where domicilio='Urquiza 283';
```

8- Vea si las inscripciones anteriores se cargaron:

```
select s.nombre,i.deporte,i.año
from socios as s
join inscriptos as i
on s.numero=i.numerosocio;
```

Ejecutar los comandos SQL

Borrar comando.

Finalizar

Otros problemas:

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

1- Elimine las tablas, si existen.

2- Cree las tablas:

```
create table libros(
codigo int unsigned auto_increment,
titulo varchar(40),
autor varchar (30),
editorial varchar (15),
primary key (codigo)
);
```

```
create table socios(
numero int unsigned auto_increment,
documento char(8) not null,
nombre varchar(30),
```

```

domicilio varchar(30),
primary key (numero)
);

create table prestamos(
codigolibro int unsigned not null,
numerosocio int not null,
fechaprestamo date not null,
fechadevolucion date,
primary key(codigolibro,fechaprestamo)
);

```

3- Ingrese algunos registros para las 3 tablas:

```

insert into socios values(20,'20222000','Juan Perez','Colon 123');
insert into socios values(31,'25222111','Juan Perez','Sucre 34');
insert into socios values(52,'25333222','Ana Maria Lopez','Avellaneda 235');
insert into socios values(82,'28333222','Luisa Duarte','San Martin 877');
insert into socios values(90,'29333222','Carlos Fuentes','Rivadavia 864');

insert into libros values (15,'Manual de 1º grado','Moreno Luis','Emece');
insert into libros values (28,'Manual de 2º grado','Moreno Luis','Emece');
insert into libros values (30,'Alicia en el pais de las maravillas','Lewis Carroll','Planeta');
insert into libros values (35,'El aleph','Borges','Emece');
insert into libros values (36,'Aprenda PHP','Molina Marcos','Planeta');
insert into libros values (40,'Cervantes y el quijote','Borges','Paidos');
insert into libros values (46,'Aprenda Java','Molina Marcos','Planeta');

insert into prestamos values(15,20,'2016-07-10','2016-07-12');
insert into prestamos values(15,31,'2016-07-12','2016-07-15');
insert into prestamos values(15,20,'2016-07-18',null);
insert into prestamos values(28,20,'2016-07-15',null);
insert into prestamos values(30,20,'2016-07-20',null);
insert into prestamos values(35,31,'2016-07-20','2016-07-22');
insert into prestamos values(36,90,'2016-07-25','2016-07-26');

```

4- El socio con número de documento "25333222" solicita en préstamos el libro con código 40. El socio no recuerda su número de socio pero si su documento. Ingrese el préstamo ubicando el número de socio a partir del documento en "socios":

```

insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select 40,numero,'2016-09-15'
from socios
where documento='25333222';

```

5- El socio llamado "Juan Perez" solicita el libro código 35; no recuerda su número de socio ni su número de documento. Intente ingresar el registro localizando el número de socio a partir del nombre:

```

insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select 35,numero,'2016-09-16'
from socios
where nombre='Juan Perez';

```

Aparece un mensaje de error indicando que la clave se duplica; porque la consulta retorna 2 registros (hay 2 socios con igual nombre), MySQL intenta insertar 2 registros iguales en todos los valores excepto en el número de documento, pero no se pueden ingresar 2 registros con el mismo código de libro en la misma fecha (clave primaria). Resuelva el problema, ingrese el préstamo colocando las condiciones necesarias para que la consulta retorne solamente una fila:

```

insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select 35,numero,'2016-09-16'
from socios
where nombre='Juan Perez' and
domicilio='Sucre 34';

```

6- Seleccione el documento y nombre de todos los morosos (una vez cada socio):

```
select distinct s.documento,s.nombre
from socios as s
join prestamos as p
on s.numero=p.numerosocio
where fechadevolucion is null;
```

7- La biblioteca necesita una tabla en la cual almacene el documento y nombre de los socios morosos. Elimine la tabla "morosos" si existe:

```
drop table if exists morosos;
```

8- Cree la tabla "morosos" a partir de la consulta realizada en el punto 6:

```
create table morosos
select distinct s.documento,s.nombre
from socios as s
join prestamos as p
on s.numero=p.numerosocio
where fechadevolucion is null;
```

9- Muestre los números de socio, documento y nombre de todos los socios que no son deudores, empleando un "left join" entre las tablas "socios" y "morosos":

```
select s.numero,s.documento,s.nombre
from socios as s
left join morosos as m
on s.documento=m.documento
where m.documento is null;
```

10- El socio con documento "20222000" solicita el préstamo de un libro. El socio no recuerda su número pero si su documento. Se desea realizar el préstamo sólo si dicho socio no se encuentra en "morosos". Realice una consulta a la tabla "socios" y "morosos" que retorne los números de socio de aquellos socios que no adeudan libros y luego busque en dicho resultado el valor del número de socio a partir del documento, si encuentra coincidencia, ingresará el registro, en caso contrario (la persona es un socio es deudor o no es socio), la inserción no se realiza:

```
insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select 15,s.documento,'2016-07-28'
from socios as s
left join morosos as m
on s.documento=m.documento
where m.documento is null
and s.documento='20222000';
```

No se realizó la carga del préstamo porque no encontró el documento en el resultado de la consulta.

11- El socio con documento "29333222" solicita el préstamo de un libro. El socio no recuerda su número pero si su documento. Se desea realizar el préstamo sólo si dicho socio no se encuentra en "morosos". Realice una consulta a la tabla "socios" y "morosos" que retorne los números de socio de aquellos socios que no adeudan libros y luego busque en dicho resultado el valor del número de socio a partir del documento:

```
insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select 36,s.numero,'2016-07-28'
from socios as s
left join morosos as m
on s.documento=m.documento
where m.documento is null
and s.documento='29333222';
```

La inserción se realizó porque el documento del socio buscado se encontró en la consulta de "no morosos".

12- Coloque fecha de devolución al libro con código "36" prestado el día "2016-07-28":

```
update prestamos
set fechadevolucion='2016-07-29'
where codigolibro=36 and
```

fechaprestamo='2016-07-28';

13- Muestre el código y nombre de todos los libros que no han sido devueltos:

```
select l.codigo,l.titulo
from libros as l
left join prestamos as p
on p.codigolibro=l.codigo
where p.fechaprestamo is not null and
fechadevolucion is null;
```

14- la biblioteca desea crear una tabla llamada "librosausentes" en la que guarde el código y título de los libros no devueltos. Elimine la tabla "librosausentes" si existe:

```
drop table if exists librosausentes;
```

15- Cree la tabla a partir de la consulta del punto 13:

```
create table librosausentes
select l.codigo,l.titulo
from libros as l
left join prestamos as p
on p.codigolibro=l.codigo
where p.fechaprestamo is not null and
fechadevolucion is null;
```

16- Realice un "left join" entre las tablas "libros" y "librosausentes" mostrando los códigos y títulos de los libros presentes:

```
select l.codigo,l.titulo
from libros as l
left join librosausentes as la
on la.codigo=l.codigo
where la.codigo is null;
```

17- El socio número 82 quiere llevar el libro código "35", ingrese el registro en "prestamos" sólo si al buscar el código en el resultado de la consulta anterior encuentra coincidencia, es decir, si el libro está presente:

```
insert into prestamos (codigolibro,numerosocio,fechaprestamo)
select l.codigo,82,'2016-09-17'
from libros as l
left join librosausentes as la
on la.codigo=l.codigo
where la.codigo is null and
l.codigo=35;
```

No se realiza la inserción del registro porque el libro está prestado.

18- El mismo socio intenta llevar el libro con código "36". Realice la misma consulta:

```
select l.codigo,82,'2016-09-17'
from libros as l
left join librosausentes as la
on la.codigo=l.codigo
where la.codigo is null and
l.codigo=36;
```

El registro se cargó porque el libro está presente.

B) Un instituto de enseñanza guarda en una tabla llamada "carreras" los datos de las carreras que dicta, en "materias" las materias de cada carrera y en "inscriptos" las inscripciones.

1- Elimine las 3 tablas, si existen.

2- Cree las tablas con las siguientes estructuras:

```
create table carreras(
codigo tinyint unsigned auto_increment,
```

```

nombre varchar(30),
primary key(codigo)
);

create table materias(
codigo tinyint unsigned auto_increment,
codigocarrera tinyint unsigned,
nombre varchar(30),
profesor varchar(30),
primary key(codigo,codigocarrera)
);

```

```

create table inscriptos(
documento char(8) not null,
codigocarrera tinyint unsigned,
codigomateria tinyint unsigned,
año year,
cuota char(1),/* si esta paga o no*/
primary key (documento,codigocarrera,codigomateria,año)
);

```

3- Ingrese algunos registros:

```

insert into carreras values(1,'Analista de sistemas');
insert into carreras values(2,'Diseñador web');

```

```

insert into materias values(1,1,'Programacion I','Alfredo Lopez');
insert into materias values(2,1,'Sistemas de datos I','Bernardo Garcia');
insert into materias values(3,1,'Ingles tecnico','Edit Torres');
insert into materias values(1,2,'Programacion basica','Alfredo Lopez');
insert into materias values(2,2,'Ingles I','Edit Torres');
insert into materias values(3,2,'Protocolos','Hector Juarez');

```

```

insert into inscriptos values('22333444',1,1,'2015','s');
insert into inscriptos values('22333444',1,2,'2015','s');
insert into inscriptos values('22333444',1,3,'2016','n');
insert into inscriptos values('23222222',1,1,'2015','s');
insert into inscriptos values('23222222',1,2,'2016','s');
insert into inscriptos values('24555666',1,1,'2015','s');
insert into inscriptos values('24555666',2,1,'2015','s');
insert into inscriptos values('25000999',1,1,'2015','s');
insert into inscriptos values('25000999',1,2,'2015','s');
insert into inscriptos values('25000999',2,1,'2016','n');
insert into inscriptos values('25000999',2,2,'2016','s');

```

4- Muestre todos los datos de la tabla "inscriptos" (sin códigos) incluyendo los nombres de las materias y carreras:

```

select i.documento,c.nombre,m.nombre,año,cuota
from inscriptos as i
join carreras as c
on c.codigo=i.codigocarrera
join materias as m
on m.codigo=i.codigomateria and
m.codigocarrera=c.codigo;

```

5- Se quiere inscribir un alumno en la materia "Programacion basica" de la carrera "Diseñador web" pero la secretaria no recuerda los códigos de las mismas. Inscriba al alumno consultando los valores de las tablas "carreras" y "materias":

```

insert into inscriptos (documento,codigocarrera,codigomateria,año,cuota)
select '30222333',c.codigo,m.codigo,'2016','s'
from carreras as c
join materias as m

```

```
on c.codigo=m.codigocarrera  
where c.nombre='Diseñador web' and  
m.nombre='Programacion basica';
```

Retornar