

## 69 - join, group by y funciones de agrupamiento.

### Problema:

Una empresa tiene registrados sus clientes en una tabla llamada "clientes", también tiene una tabla "provincias" donde registra los nombres de las provincias de las cuales son oriundos los clientes.

1- Elimine la tabla "clientes" y "provincias", si existen:  
drop table if exists clientes, provincias;

2- Créelas con las siguientes estructuras:

```
create table clientes (  
  codigo int unsigned auto_increment,  
  nombre varchar(30) not null,  
  domicilio varchar(30),  
  ciudad varchar(20),  
  codigoProvincia tinyint unsigned,  
  telefono varchar(11),  
  primary key(codigo)  
);
```

```
create table provincias(  
  codigo tinyint unsigned auto_increment,  
  nombre varchar(20),  
  primary key (codigo)  
);
```

3- Ingrese algunos registros para ambas tablas:

```
insert into provincias (nombre)  
values('Cordoba');  
insert into provincias (nombre)  
values('Santa Fe');  
insert into provincias (nombre)  
values('Corrientes');  
insert into provincias (nombre)  
values('Misiones');  
insert into provincias (nombre)  
values('Salta');  
insert into provincias (nombre)  
values('Buenos Aires');  
insert into provincias (nombre)  
values('Neuquen');
```

```
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Lopez Marcos', 'Colon 111', 'Córdoba',1,null);  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Perez Ana', 'San Martin 222', 'Cruz del Eje',1,'4578585');  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Garcia Juan', 'Rivadavia 333', 'Villa Maria',1,'4578445');  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Perez Luis', 'Sarmiento 444', 'Rosario',2,null);  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Pereyra Lucas', 'San Martin 555', 'Cruz del Eje',1,'4253685');  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Gomez Ines', 'San Martin 666', 'Santa Fe',2,'0345252525');  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)  
values ('Torres Fabiola', 'Alem 777', 'Villa del Rosario',1,'4554455');  
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)
```

```
values ('Lopez Carlos', 'Irigoyen 888', 'Cruz del Eje',1,null);
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)
values ('Ramos Betina', 'San Martin 999', 'Cordoba',1,'4223366');
insert into clientes (nombre,domicilio,ciudad,codigoProvincia,telefono)
values ('Lopez Lucas', 'San Martin 1010', 'Posadas',4,'0457858745');
```

4- Agrupe por nombre de provincia y cuente la cantidad de clientes por provincia usando un "join":

```
select p.nombre,
count(c.codigoprovincia) as 'cant. clientes'
from provincias as p
join clientes as c
on p.codigo=c.codigoprovincia
group by p.nombre;
```

sólo aparecen las provincias en las cuales tenemos clientes.

5- Agrupe por nombre de provincia y cuente la cantidad de clientes por provincia usando un "left join":

```
select p.nombre,
count(c.codigoprovincia) as 'cant. clientes'
from provincias as p
left join clientes as c
on p.codigo=c.codigoprovincia
group by p.nombre;
```

Muestra todas las provincias.

6- Agrupe por nombre de provincia y muestre la cantidad de clientes por provincia usando un "join" de las provincias en las cuales tenemos 2 o más clientes:

```
select p.nombre,
count(c.codigoprovincia) as 'cant. clientes'
from provincias as p
join clientes as c
on p.codigo=c.codigoprovincia
group by p.nombre
having count(c.codigoprovincia)>=2;
```

Ejecutar los comandos SQL

Borrar comando.

Finalizar

### Otros problemas:

A) Un comercio que tiene un stand en una feria registra en una tabla llamada "visitantes" algunos datos de las personas que visitan o compran en su stand para luego enviarle publicidad de sus productos.

1- Elimine las tablas "visitantes" y "ciudades", si existen.

2- Créelas con las siguientes estructuras:

```
create table visitantes(  
  nombre varchar(30),  
  edad tinyint unsigned,  
  sexo char(1),  
  domicilio varchar(30),  
  codigociudad tinyint unsigned not null,  
  telefono varchar(11),  
  montocompra decimal(6,2) unsigned  
);
```

```
create table ciudades(  
  codigo tinyint unsigned auto_increment,  
  nombre varchar(20),  
  primary key (codigo)  
);
```

3- Ingrese algunos registros:

```
insert into ciudades (nombre)  
values('Cordoba');  
insert into ciudades (nombre)  
values('Alta Gracia');  
insert into ciudades (nombre)  
values('Villa Dolores');  
insert into ciudades (nombre)  
values('Carlos Paz');
```

```
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Susana Molina', 28,'f','Colon 123',1,null,45.50);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Marcela Mercado',36,'f','Avellaneda 345',1,'4545454',0);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Alberto Garcia',35,'m','Gral. Paz 123',2,'03547123456',25);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Teresa Garcia',33,'f','Gral. Paz 123',2,'03547123456',0);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Roberto Perez',45,'m','Urquiza 335',1,'4123456',33.20);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Marina Torres',22,'f','Colon 222',3,'03544112233',25);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Julieta Gomez',24,'f','San Martin 333',2,'03547121212',53.50);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Roxana Lopez',20,'f','Triunvirato 345',2,null,0);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Liliana Garcia',50,'f','Paso 999',1,'4588778',48);  
insert into visitantes (nombre,edad, sexo,domicilio,codigociudad,telefono,montocompra)  
values ('Juan Torres',43,'m','Sarmiento 876',1,'4988778',15.30);
```

4- Muestre la cantidad de visitantes agrupados por nombre de la ciudad:

```
select c.nombre,count(v.codigociudad)  
from ciudades as c  
left join visitantes as v  
on c.codigo=v.codigociudad  
group by c.nombre;
```

5- Muestre la cantidad de visitantes que hicieron alguna compra, agrupados por nombre de la ciudad:

```
select c.nombre,count(v.codigociudad)  
from ciudades as c  
join visitantes as v  
on c.codigo=v.codigociudad  
where v.montocompra>0
```

group by c.nombre;

6- Muestre la suma de las compras y el promedio de las mismas, agrupados por ciudad y sexo:

```
select c.nombre,sexo,sum(montocompra) as 'total',  
avg(montocompra) as 'promedio'  
from ciudades as c  
join visitantes as v  
on c.codigo=v.codigociudad  
group by c.nombre,sexo;
```

B) Una inmobiliaria que alquila departamentos guarda la información de los mismos en una tabla llamada "departamentos" y "barrios".

1- Elimine las tablas si existen.

2- Cree las tablas con las siguientes estructuras:

```
create table departamentos(  
edificio varchar(30),  
domicilio varchar(30) not null,  
piso char(1) not null,  
numerodpto char(2) not null,  
detalles varchar(200),  
codigobarrio tinyint unsigned,  
precio decimal(6,2) unsigned,  
primary key (edificio,piso,numerodpto)  
);
```

```
create table barrios(  
codigo tinyint unsigned auto_increment,  
nombre varchar(30),  
primary key(codigo)  
);
```

3- Ingrese los siguientes registros:

```
insert into barrios (nombre) values ('Centro');  
insert into barrios (nombre) values ('Alberdi');  
insert into barrios (nombre) values ('Gral. Paz');  
insert into barrios (nombre) values ('Pueyrredon');
```

```
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Avellaneda','Avellaneda 86','1','1',1,400.50);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Avellaneda','Avellaneda 86','1','2',1,400.50);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Avellaneda','Avellaneda 86','2','1',1,400.50);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Bolívar','Sarmiento 1203','1','1',3,500);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Centauro I','Peru 456','1','A',4,300);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Centauro I','Peru 456','2','C',4,350);  
insert into departamentos (edificio,domicilio,piso,numerodpto,codigobarrio,precio)  
values('Paris','Urquiza 364','1','12',1,600);
```

4- Muestre todos los departamentos incluido el nombre del barrio:

```
select edificio,domicilio,piso,numerodpto,nombre,precio  
from departamentos as d  
join barrios as b  
on d.codigobarrio=b.codigo;
```

5- Muestre la cantidad de departamentos por edificio con el nombre del barrio:

```
select edificio,nombre,count(*)
from departamentos as d
join barrios as b
on d.codigobarrio=b.codigo
group by edificio;
```

6- Muestre el promedio de los precios de los departamentos agrupados por barrio:

```
select nombre,avg(precio)
from departamentos as d
join barrios as b
on d.codigobarrio=b.codigo
group by nombre;
```

7- Muestre el promedio de los precios de los departamentos agrupados por barrio teniendo en cuenta todos los barrios, incluso aquellos en los cuales no hay departamentos disponibles:

```
select nombre,avg(precio)
from barrios as b
left join departamentos as d
on d.codigobarrio=b.codigo
group by nombre;
```

C) Un video club que alquila películas en video guarda información de sus películas en alquiler y los alquileres en las tabla "peliculas" y "alquileres" respectivamente.

1- Elimine las tablas si existen.

2- Créelas con las siguientes estructuras:

```
create table peliculas (
codigo smallint unsigned auto_increment,
titulo varchar(50) not null,
actores varchar(40),
duracion tinyint unsigned,
primary key (codigo)
);
```

```
create table alquileres(
codigopelicula smallint unsigned not null,
socio varchar(30) not null,
fechaprestamo date not null,
fechadevolucion date,
primary key (codigopelicula,fechaprestamo)
);
```

3- Ingrese los siguientes registros para las 2 tablas.

```
insert into peliculas (titulo,actores,duracion)
values('Elsa y Fred','China Zorrilla',90);
insert into peliculas (titulo,actores,duracion)
values('Mision imposible','Tom Cruise',120);
insert into peliculas (titulo,actores,duracion)
values('Mision imposible 2','Tom Cruise',180);
insert into peliculas (titulo,actores,duracion)
values('Harry Potter y la piedra filosofal','Daniel H.',120);
insert into peliculas (titulo,actores,duracion)
values('Harry Potter y la camara secreta','Daniel H.',150);
```

```
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(1,'Juan Lopez','2016-07-02');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(2,'Juan Lopez','2016-07-02');
```

```

insert into alquileres (codigopelicula,socio,fechaprestamo)
values(3,'Juan Lopez','2016-07-12');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(1,'Luis Molina','2016-08-02');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(3,'Luis Molina','2016-08-12');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(4,'Luis Molina','2016-08-02');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(1,'Andrea Torres','2016-09-02');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(2,'Andrea Torres','2016-08-02');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(3,'Andrea Torres','2016-08-15');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(4,'Andrea Torres','2016-08-22');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(4,'Juan Lopez','2016-08-25');
insert into alquileres (codigopelicula,socio,fechaprestamo)
values(1,'Andrea Torres','2016-08-25');

```

4- Muestre toda la información de los "alquileres" (nombre de la película, nombre del socio, fecha de préstamo y de devolución):

```

select titulo,socio,fechaprestamo,fechadevolucion from alquileres as a
join peliculas as p
on a.codigopelicula=p.codigo;

```

5- Muestre la cantidad de veces que se alquiló cada película:

```

select p.titulo,count(*) from peliculas as p
join alquileres as a
on p.codigo=a.codigopelicula
group by p.titulo;

```

6- Muestre la cantidad de películas que alquiló cada socio:

```

select socio,count(a.codigopelicula) from alquileres as a
group by socio;

```

7- Muestre la cantidad de películas DISTINTAS que alquiló cada socio:

```

select socio,count(distinct a.codigopelicula) from alquileres as a
group by socio;

```

8- Muestre la cantidad de películas alquiladas por mes por cada socio ordenado por mes:

```

select socio,
monthname(a.fechaprestamo) as mes,
count(a.codigopelicula)
from alquileres as a
group by socio, mes
order by mes;

```

[Retornar](#)