

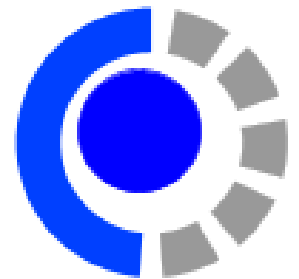
CONCEPTOS DE BASES DE DATOS

SQL - Structured Query Language

UNIDAD V



Departamento Ingeniería de Sistemas
Facultad de Informática
Universidad Nacional del Comahue



SQL – Agrupamiento

2

- La cláusula **Group by** es usada para mostrar las filas y columnas agrupadas por columnas selectivas.
- Puede usarse para ver las funciones agregadas, como count().
 - ▣ Ejemplo: Se desea saber cuantos clientes hay por localidad.

```
SELECT COUNT(*), localidad  
FROM clientes  
GROUP BY localidad
```

SQL – Agrupamiento

3

clientes.clientes	
idCliente	: int(11)
NroDNI	: int(11)
FechaAlta	: date
Nombre	: varchar(100)
FechaUltimaCompra	: date
idProvincia	: int(11)
Localidad	: varchar(50)
Domicilio	: varchar(256)
TEFijo	: varchar(20)
FAX	: varchar(20)
TEMovil	: varchar(20)
Deuda	: float
VIP	: int(11)

**SELECT COUNT(*),
Localidad FROM clientes
GROUP BY Localidad**



COUNT(*)	localidad
4	Allen
6	Bahia Blanca
16	CABA
6	Caleta Olivia
5	Centenario
4	Cipolletti
3	Comodoro Rivadavia
5	Córdoba
9	Corrientes
10	Eldorado
13	Formosa
5	Gualectuaychu
7	Jujuy
6	La Plata
9	La Rioja
8	Mendoza
2	Merlo

El campo que se use para agrupar debe ser un campo incluido en el SELECT.

SQL - Ordenamiento

4

- A menudo es más fácil examinar la salida de una consulta cuando las filas se ordenan de algún modo significativo.
- Para ordenar un resultado, se usa la cláusula **ORDER BY** luego de la cláusula **WHERE**.
- Ejemplo: listar todos los cliente de la localidad de Neuquén ordenados por nombre:

```
SELECT *
```

```
FROM clientes WHERE Localidad = 'Neuquén'
```

```
ORDER BY Nombre
```

El campo que se use para ordenar debe ser un campo incluido en el SELECT.

SQL - Ordenamiento

5

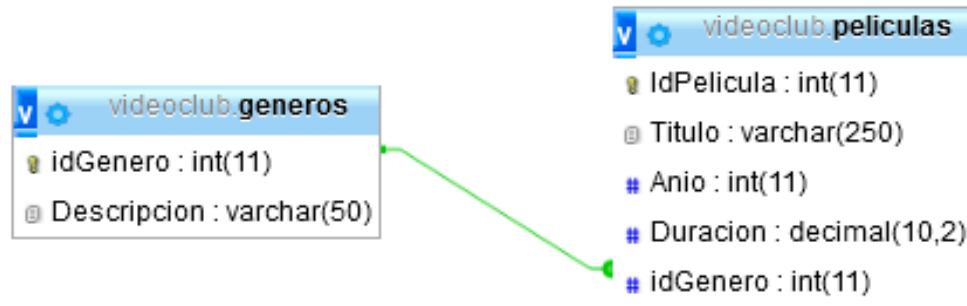
idCliente	NroDNI	FechaAlta	Nombre ▲	FechaUltima	idProvincia	Localidad	Domicilio	TEFijo	FAX	TEMovil	Deuda	V
39	16052263	2006-04-03	Hector Omar Pascal	2012-07-26	15	Neuquén	Rio Carcaraña 574	661736184	1971887523	1771728991	66376.1	
1	43535633	2010-03-06	Martin Jose Barreto Bastidas	2012-04-14	15	Neuquén	PRINGLES 1500	2257359165	2847183555	1581961819	59077.5	
50	93036393	2012-07-05	Nora Mabel Campos Gutierrez	2012-09-08	15	Neuquén	Socomba 365	1116556152	2258470298	5193149084	90906.8	
36	92507283	2012-04-17	Raul Contreras Contreras	2012-08-26	15	Neuquén	complejo chocon 0	2810044272	6482432943	249874149	30378	

El campo que se use para ordenar debe ser un campo incluido en el SELECT.

SQL - Combinación de tablas Relacionadas

6

- Las vinculaciones entre tablas se realizan mediante la cláusula **INNER JOIN** que combina registros de dos tablas siempre que haya una relación entre ellas.



- Ejemplo:
 - La tabla películas hace referencia a la tabla géneros a través de la clave foránea “idGenero”.
 - Se desea saber a qué género pertenece cada película.

SQL - Combinación de tablas Relacionadas

7

- Se puede utilizar una operación **INNER JOIN** en cualquier cláusula **FROM**. Esto crea una combinación por equivalencia. Su sintaxis es:

SELECT campos

FROM tb1 **INNER JOIN** tb2 **ON**
tb1.campo1 = tb2.campo2

- Ejemplo:

SELECT Titulo, Descripcion **FROM** peliculas
INNER JOIN generos **ON** peliculas.idGenero =
generos.idGenero

Titulo	Descripcion
El sexto sentido	Drama
Cuento de invierno	Romance
Identidad secreta	Drama
Un Angel Enamorado	Romance
Oculus	Terror



SQL - Combinación de tablas Relacionadas

8

- Puede haber mas de un INNER JOIN en una consulta:
SELECT lista_campos **FROM** tab1
INNER JOIN tab2 **ON** join_condition1
INNER JOIN tab3 **ON** join_condition2
...
WHERE where_conditions;



SQL - Combinación de tablas Relacionadas

9

- Ejemplo: Queremos saber el titulo, el genero y en que estado están las películas.

SELECT peliculas.Titulo, generos.Descripcion,
estados.Descripcion

FROM peliculas

INNER JOIN generos **ON** peliculas.idGenero =
generos.idGenero

INNER JOIN estados **ON** peliculas.idEstado =
estados.idEstado

Titulo	Descripcion	Descripcion
El sexto sentido	Drama	Alquilada
Cuento de invierno	Romance	Disponible

SQL - Alias

10

- En el ejemplo visto anteriormente, que sucede con el campo descripción en el resultado de la consulta?
- Es posible asignar un alias a cualquiera de las expresiones en el SELECT. Esto se puede hacer usando la palabra **AS**.
- También se pueden usar alias para referenciar a las tablas involucradas en las consultas
- Ejemplo:

```
SELECT P.Titulo, G.Descripcion AS Genero, E.Descripcion AS Estado  
FROM peliculas P  
INNER JOIN generos G ON P.idGenero = G.idGenero  
INNER JOIN estados E ON P.idEstado = E.idEstado
```

Titulo	Genero	Estado
El sexto sentido	Drama	Alquilada
Cuento de invierno	Romance	Disponible

Material

11

- Manual Mysql

- <https://dev.mysql.com/doc/refman/5.0/es/>