

Universidad del Quindío - Ingeniería de Sistemas - Leonardo Hernández R.
 leonardohernandez@telesat.com.co www.geocities.com/leoher314
 Mar.3/2002, Sep.14/2002, Mar.17/2003

Tutorial de MySQL

----- Lección IV Funciones de Agregación -----

4.1. Introducción

Las funciones de agregación nos permiten sumar, contar, hallar promedios, valores máximos y mínimos y desviaciones estándar. Tienen las siguientes características:

- Trabajan con conjuntos de registros y no con datos aislados. Esta es una diferencia importante con otras funciones como el seno, la raíz cuadrada etc.
- Pueden ser aplicadas sobre toda la tabla o por grupos de registros. Por ejemplo se puede hallar la suma de los sueldos de una empresa, pero también, la suma de los sueldos por cada dependencia.
- Para realizar los cálculos mencionados, no se necesita especificar ciclos ni inicializaciones

Las funciones de agregación son:

count	contar
sum	suma
min	mínimo
max	máximo
avg	promedio
std	desviación estándar en MySQL
stddev	desviación estándar en Oracle

4.2. Tabla de ejemplo para esta lección

Para esta sección usaremos la misma tabla de la lección anterior, la tabla Empleado, la cual se supone ya creada con muchísimos registros:

```
Empleado ( idEmpleado, nombre, apellido, sexo,
           estadoCivil, fechaNacimiento, dependencia, sueldo)
```

Recordemos que el atributo *idEmpleado* es un número consecutivo que identifica el empleado. El sexo puede ser M o F. El estado civil se registra según la siguiente convención:

s	soltero
c	casado
u	unión libre
v	viudo
r	religioso

El atributo *dependencia* corresponde al nombre de la dependencia donde labora el empleado.


```
mysql> select count(*) from Empleado where sexo ='f';
```

e) Número de empleados por estado civil.

```
mysql> select estadoCivil, count(*) from Empleado
-> group by estadoCivil;
```

estadoCivil	count(*)
c	5
s	2
u	3
v	1

El resultado es una tabla de dos columnas ya que la lista de atributos del select es de dos atributos: *estadoCivil* y *count(*)* que representan respectivamente el estado civil y el número de empleados por cada estado civil.

En este ejemplo, no se produce un solo resultado global, sino varios parciales, uno por cada estado civil. Dicho de otra manera la tabla se subdivide en grupos, uno por cada estado civil: el grupo de los registros de los casados, el grupo de los registros de los solteros etc. Posteriormente la función de agregación se aplica a cada grupo.

Cuando la función de agregación no va a producir un solo resultado global sino varios parciales, se debe utilizar la cláusula *group by* seguida de la lista de los atributos que se utilizarán para la agrupación.

Un secreto para no equivocarse en el uso de la cláusula *group by* es el siguiente:

Generalmente, cuando una función de agregación, va acompañada de otros atributos, debe usarse la cláusula *group by* con exactamente estos atributos. Observe en negrilla en el ejemplo, que el atributo que acompaña a la función *count* es exactamente el atributo de la cláusula *group by*, el atributo *estadoCivil*.

Ya se puede apreciar el poder del lenguaje SQL, una instrucción como la anterior reemplaza a varias instrucciones estándar de un lenguaje procedimental, como C++;

Ahora supongamos que no necesitamos todos los renglones de la última salida, que solo requerimos aquellos renglones en que el contador es mayor que 2. En este caso se utiliza la cláusula *having*:

```
mysql> select estadoCivil, count(*) from Empleado
-> group by estadoCivil
-> having count(*) > 2;
```

estadoCivil	count(*)
c	5
u	3

No confunda la cláusula *where* con la cláusula *having*. La primera selecciona registros de la tabla de entrada del *select*. La segunda selecciona renglones de

una salida con funciones de agregación. Dicho de otro modo, tiene sentido usar la cláusula *having* si se ha usado una o varias funciones de agregación y la cláusula *group by*.

f) Número de empleados por sexo - estado civil.

```
mysql> select sexo, estadoCivil, count(*) from Empleado
        group by sexo, estadoCivil;
```

sexo	estadoCivil	count(*)
f	c	3
f	s	2
f	v	1
m	c	2
m	s	1
m	u	2

Note como se ha agrupado por dos atributos y no por uno. De nuevo la lista de atributos que acompañan a la función de agregación es la misma lista de atributos de la cláusula *group by*.

4.4. Otras funciones de agregación

La función de agregación *sum* se utiliza de manera similar a la función *count*. A continuación se dan ejemplos de salidas con su correspondiente instrucción SQL:

a) Suma de los sueldos de todos los empleados.

```
mysql> select sum(sueldo) from Empleado;
```

b) Suma de los sueldos de las mujeres casadas:

```
mysql> select sum(sueldo) from Empleado
-> where sexo='m' and estadoCivil='c';
```

c) Suma de los sueldos por dependencia.

```
mysql> select dependencia, sum(sueldo) from Empleado
-> group by dependencia;
```

d) Suma de los sueldos por dependencia - sexo:

```
mysql> select dependencia, sexo, sum(sueldo) from Empleado
        group by dependencia, sexo;
```

e) Suma de los sueldos por dependencia, pero solo las dependencias donde esta suma sea al menos \$2'000.000.

```
mysql> select dependencia, sum(sueldo) from Empleado
        group by dependencia
        having sum(sueldo) >= 2000000;
```

Recuerde que *having* se utiliza cuando no se desean todos los renglones de una salida con funciones de agregación, sino que interesan solo ciertos renglones.

Las funciones *min*, *max*, *avg* y *std* (o *stddev*) se utilizan de manera similar.

4.5. Ejercicios

Elabore consultas SQL(una por cada numeral) para hallar:

5. El número de empleados por cada estado civil.
10. El número de empleados casados o en unión libre, por sexo.
15. El número de empleados con fecha de nacimiento conocida, por sexo.
20. El número de empleados nacidos cada año.
25. El numero de empleados nacidos cada año, pero sólo los años cuando hayan nacido 2 o más empleados.
30. El número de empleados por año de nacimiento - sexo, pero solo cuando este número sea mayor que 1.
35. El promedio se sueldo de los empleados.
40. El máximo sueldo ganado por un hombre casado.
45. El promedio de sueldo de las mujeres que trabajan en la biblioteca.
50. El mínimo sueldo ganado por los empleados casados o en unión libre.
55. El promedio de sueldo por estado civil.
60. El máximo sueldo por sexo - estado civil.
65. El mínimo sueldo de cada dependencia - sexo
70. El sueldo máximo, mínimo y promedio por dependencia - estado civil.
75. El promedio de sueldo por dependencia, pero solo las dependencias con sueldopromedio superior a \$1'000.000.
80. El sueldo máximo y el sueldo mínimo de las dependencias de más de 2 empleados.
85. El número de empleados y la suma de los sueldos de las dependencias donde por lo menos un empleado gana más de \$1'000.000.

4.6. Respuestas:

5.

```
select estadoCivil, count(*) from Empleado group by estadoCivil;
```
10.

```
select sexo, count(*) from Empleado
where estadoCivil='c' or estadoCivil='u'
group by sexo;
```
15.

```
select sexo, count(*) from Empleado
where fechaNacimiento is not null
group by sexo;
```
- o

```
select sexo, count(fechaNacimiento) from Empleado
group by sexo;
```
20.

```
select year(fechaNacimiento), count(*) from Empleado
group by year(fechaNacimiento);
```
25.

```
select year(fechaNacimiento), count(*) from Empleado
group by year(fechaNacimiento) having count(*) > 1;
```
30.

```
select year(fechaNacimiento), sexo, count(*) from Empleado
group by year(fechaNacimiento), sexo
having count(*) > 1;
```
35.

```
select avg(sueldo) from Empleado;
```
40.

```
select max(sueldo) from Empleado
```

```
where sexo = 'm' and estadoCivil = 'c';
```

45.

```
select avg(sueldo) from Empleado
where sexo = 'f' and dependencia = 'biblioteca';
```
50.

```
select min(sueldo) from Empleado
where estadoCivil = 'c' or estadoCivil = 'u';
```
55.

```
select estadoCivil, avg(sueldo) from Empleado
group by estadoCivil;
```
60.

```
select sexo, estadoCivil, max(sueldo) from Empleado
group by sexo, estadoCivil;
```
65.

```
select dependencia, sexo, min(sueldo) from Empleado
group by dependencia, sexo;
```
70.

```
select dependencia, estadoCivil, max(sueldo), min(sueldo), avg(sueldo)
from Empleado
group by dependencia, estadoCivil;
```
75.

```
select dependencia, avg(sueldo) from Empleado
group by dependencia having avg(sueldo) > 1000000;
```
80.

```
select dependencia, max(sueldo), min(sueldo) from Empleado
group by dependencia having count(*) > 2;
```
85.

```
select dependencia, count(*), sum(sueldo) from Empleado
group by dependencia having max(sueldo) > 1000000;
```