



[Inicio](#) > [SQL](#) > [Lecciones SQL](#) >

P08 Agregación

CONTENIDOS

- 1 Group by
 - 1.1 Where y group by
 - 1.2 Atributos no clave
 - 1.3 Ordenación de la salida
 - 1.4 Having

Group by

Recordemos que el uso de funciones de agregados nos permite realizar cálculos sobre la totalidad de filas que cumplen una determinada condición (o sobre todas las filas de la tabla)

Cantidad de créditos que se imparten

```
select sum(creditos) credsImp
from imparte i, asignaturas a
where i.asignatura = a.codigo
```

credsImp
18.0

Sin embargo, es muy habitual la necesidad de realizar ese cálculo no de forma global sino particularizando por algún criterio.

Cantidad de créditos que imparte **cada profesor**.

```
select sum(creditos) credsImp
from imparte i, asignaturas a
where i.asignatura = a.codigo
group by dni
```

credsImp

12.0
6.0

En realidad, se entenderá mejor si introducimos más información en la salida

```
select dni, sum(creditos) credsImp
from imparte i, asignaturas a
where i.asignatura = a.codigo
group by dni
```

dni	credsImp
21111222	12.0
21333444	6.0

El criterio definido en el group by (dni), establece los distintos grupos de filas sobre los que se va a realizar la suma. Digamos que el sistema busca cada uno de los valores distintos de dni dentro de la tabla imparte y que, para todas las filas que comparten cada uno de esos valores hace una suma de créditos.

A partir de aquí ya podemos añadir cualquier información que consideremos adecuada.

Nombre de los profesores y créditos que imparten.

```
select nombre, sum(creditos) credsPorProf
from imparte i, asignaturas a, profesores p
where i.asignatura = a.codigo
and p.dni = i.dni
group by p.dni
```

SQL Error: 'ejemplo.p.nombre' isn't in GROUP BY

El error en la interpretación de la orden anterior viene dado por que **es obligatorio** poner en el group by todas las columnas no calculadas que vayan a salir en el resultado.

Al contrario, no es necesario que todas las columnas especificadas en el group by aparezcan en la salida.

```
select nombre, sum(creditos) credsPorProf
from imparte i, asignaturas a, profesores p
where i.asignatura = a.codigo
and p.dni = i.dni
group by p.dni, nombre
```

nombre	credsPorProf
EVA GOMEZ	12.0
RAFAEL ROMERO	6.0

Aunque con el "nombre" bastaba, hemos puesto también el "DNI" por prevención, podría darse el caso de tener 2 profesores distintos que se llamaran igual.

Where y group by

Cuando se utiliza la cláusula where, aparte de enlazar tablas por columnas comunes, como PROFESORES e IMPARTE por profesores.dni e imparte.dni respectivamente, se puede utilizar para eliminar ciertas filas del cálculo.

Así, si queremos calcular cuantas asignaturas imparte cada profesor sin contar FBD, escribiremos la siguiente sentencia.:

```
select nombre, count(*)
from profesores p, imparte i
where p.dni = i.dni
      and asignatura != 'FBD'
group by p.dni, nombre;
```

nombre	count(*)
EVA GOMEZ	1
RAFAEL ROMERO	1

Para entendernos, "primero" se ejecuta el select-from-where sin cálculos y "después", sobre las filas resultantes, se realizan los cálculos según el criterio de agrupación del *group by*.

Atributos no clave

Hemos estado introduciendo la clave primaria de profesores en los group by para prevenir cálculos, en principio, erróneos. Los únicos atributos que aseguran la identificación de una tupla respecto de las demás son los que forman la clave primaria (en general, clave candidata).

Así, si la clave primaria de una tabla de personas (profesores o alumnos, por ejemplo) es el D.N.I., asumimos que no habrá duplicados en este atributo. No obstante, el nombre no será clave y, por lo tanto, admite duplicados. Es decir, es perfectamente posible encontrar a dos personas distintas que se llamen igual.

Nombre del coordinador y cuántas asignaturas coordina.

```
select nombre, count(*) coordina
from asignaturas a, coordinadores c
where c.asig = a.codigo
group by nombre;
```

ombre	coordina
AGAPITO CIFUENTES	2
ROMUALDO GOMEZ	1

DNI y nombre del coordinador y cuántas asignaturas coordina.

```
select dni, nombre, count(*) coordina
from asignaturas a, coordinadores c
where c.asig = a.codigo
group by dni, nombre;
```

dni	nombre	coordina
10111222	AGAPITO CIFUENTES	1
55777666	AGAPITO CIFUENTES	1
66555444	ROMUALDO GOMEZ	1

Parece claro que la primera solución no es correcta, más si cabe teniendo en cuenta el diseño de la tabla y el significado de sus datos (un coordinador no puede coordinar más de una asignatura).

Ordenación de la salida

Se puede ordenar la salida utilizando las columnas calculadas. Todas estas órdenes obtienen la misma salida:

- utilizando la propia expresión

```
select nombre, count( * ) asignaturas
from profesores p, imparte i
where p.dni=i.dni
group by p.dni, nombre
order by count( * );
```

- utilizando la etiqueta de columna

```
select nombre, count( * ) asignaturas
from profesores p, imparte i
where p.dni=i.dni
```

```
group by p.dni, nombre  
order by asignaturas;
```

- utilizando el orden de la columna

```
select nombre, count( * ) asignaturas  
from profesores p, imparte i  
where p.dni=i.dni  
group by p.dni, nombre  
order by 2;
```

nombre	asignaturas
RAFAEL ROMERO	1
EVA GOMEZ	2

Having

Una vez que se obtiene información calculada sobre grupos de filas mediante el *group by*, se pueden filtrar estos resultados y mostrar sólo aquellos que nos interesen mediante la cláusula *having*. *Having* es a *group by* lo mismo que *where* a *select-from*.

Where filtra los datos almacenados en la tabla y *having* la información calculada. Si la cláusula *where* selecciona filas, la cláusula *having* selecciona grupos; si en la *where* la condición que se especifica afecta a las tuplas de toda la tabla, el *group by* efectúa cálculos en función de esa selección previa y da como resultado una tabla con la información calculada para cada grupo dentro de esa selección. Sobre esta última el *having* eliminaría aquellas tuplas que no cumplen la condición.

Podemos entender el proceso de una consulta con *group by-having* como la ejecución previa de la parte *select-from-where* y, a partir de este resultado intermedio, la obtención de la información calculada con *group by* y, finalmente, el filtrado con *having*.

¿Cuántas asignaturas imparte cada profesor?

```
select p.dni, nombre, count(*) asignaturas  
from profesores p, imparte i  
where p.dni = i.dni  
group by p.dni, nombre
```

dni	nombre	asignaturas
21111222	EVA GOMEZ	2
21333444	RAFAEL ROMERO	1

¿Cuántas asignaturas imparten los profesores con 2 o más asignaturas?

```
select p.dni, nombre, count(*) asignaturas
from profesores p, imparte i
where p.dni = i.dni
group by p.dni, nombre
having count(*) >= 2
```

dni	nombre	asignaturas
21111222	EVA GOMEZ	2

Having no permite utilizar la etiqueta de las columnas calculadas, en este caso "asignaturas", ni tampoco el orden de la columna que interpretaría como un número entero.

Sin tener en cuenta la asignatura FBD, ¿cuántas asignaturas imparten los profesores con 2 o más asignaturas?

```
select p.dni, nombre, count(*) asignaturas
from profesores p, imparte i
where p.dni = i.dni
and asignatura != 'FBD'
group by p.dni, nombre
having count(*) >= 2
```

Ninguna fila seleccionada

Paso a paso, en la consulta anterior:

1. *where* ha eliminado las tuplas de *imparte* de código de asignatura FBD.
2. *group by* calcula, para cada profesor, el número de asignaturas que imparte. Puesto que no contamos FBD, Eva Gómez solo imparte, igual que Rafael Romero, una única asignatura.
3. *having* elimina del resultado del paso anterior todas aquellas tuplas con un valor en la cuenta de filas menor que 2. El resultado es vacío puesto que ninguno de los grupos supera la condición.

La condición del *having* no es necesariamente sobre el cálculo a mostrar, como se puede ver en el siguiente ejemplo.

¿Cuántas asignaturas imparte cada profesor en el caso de que la suma de créditos sea mayor que 5?

```
select p.dni, nombre, count(*) asignaturas
from profesores p, imparte i, asignaturas a
where p.dni = i.dni and a.codigo = i.asignatura
group by p.dni, nombre
having sum(creditos) > 5
```

dni	nombre	asignaturas
21111222	EVA GOMEZ	2
21333444	RAFAEL ROMERO	1



FBDdocs por [BDgite](#) se encuentra bajo una [Licencia Creative Commons Atribución-CompartirIgual 3.0 Unported](#). Basada en una obra en <http://fbddocs.dlsi.ua.es>. Permisos que vayan más allá de lo cubierto por esta licencia pueden encontrarse en <http://fbddocs.dlsi.ua.es/autores>.

BDgite (GITE-11014-UA),
Departamento de Lenguajes y Sistemas Informáticos,
Universidad de Alicante

[Iniciar sesión](#) | [Informar de uso inadecuado](#) | [Imprimir página](#) | Con la tecnología de [Google Sites](#)