

# **Bases de Datos**

## **Unidad 5: Realización de consultas avanzadas Sesión 4**

**Consultas de resumen o agregado**

**Agrupamiento de registros**

Las **consultas de resumen o de agregado** permiten realizar cálculos de resumen o de grupo sobre las filas que tienen un valor igual en una o varias columnas.

Para realizar estos cálculos se usan las **funciones de agregado**.

- **Count(*expresión o columna*)**: Cuenta cuantas filas hay con la expresión o columna que no estén a valor nulo. Si en el argumento de la función escribimos \*, se cuentan cuantas filas hay en la consulta. Si la expresión o columna vale **null**, no se cuenta.
- **Sum(*expresión o columna*)**: Calcula la suma de los valores numéricos indicados en el argumento. Si en la expresión o columna hay **null**, no se tiene en cuenta para la suma.
- **Min(*expresión o columna*)**: Obtiene el valor mínimo del argumento indicado.
- **Max(*expresión o columna*)**: Obtiene el valor máximo del argumento indicado.
- **Avg(*expresión o columna*)**: Obtiene la media aritmética del argumento indicado. No considera los valores nulos para el cálculo de la media.
- **Group\_concat(*expresión o columna*)**: Obtiene la concatenación de todos los valores que se obtendrían en la consulta. No considera los valores nulos para la concatenación.

*Ejemplo: Obtener cuantos contratos se han realizado:*

```
SELECT count(*) FROM contratos;
```

*Ejemplo: Obtener cuantos contratos realizados han finalizado.*

```
SELECT count(ffin) FROM contratos;
```

*Ejemplo: Obtener cuantos automóviles hay.*

```
SELECT count(*) FROM automoviles;
```

*Ejemplo: Obtener de cuantas marcas hay coches.*

```
SELECT count(marca) FROM automoviles;
```

¿Es correcta esta última consulta?

## Consultas de resumen

*Ejemplo: Obtener la media de kilómetros realizados en los contratos finalizados, el máximo kilometraje realizado y el mínimo.*

**SELECT avg(kfin-kini), max(kfin-kini), min(kfin-kini) FROM contratos;**

avg(kfin-kini)	max(kfin-kini)	min(kfin-kini)
1634.0667	5671	43

¿Es coherente la consulta mostrada con el ejemplo propuesto?  
¿Qué tendría que incluir la consulta?

## Consultas de resumen

*Ejemplo: Obtener una cadena de caracteres concatenación de los nombres de todos los clientes de Toledo.*

```
SELECT group_concat(nombre) FROM clientes WHERE localidad='toledo';
```

group_concat(nombre)
Fanny,Noelia,Natalia,Ana Belén

## Consultas de resumen

*Ejemplo: Obtener la suma total de kilómetros realizados en contratos finalizados por clientes de Madrid.*

```
SELECT sum(kfin-kini) FROM contratos INNER JOIN clientes ON dnicliente=dni  
WHERE localidad='madrid';
```

sum(kfin-kini)
22293

## Agrupamiento de registros

La cláusula **GROUP BY** permite agrupar varias filas de una consulta por una o varias expresiones. Todos los valores repetidos de las expresiones agrupadas, se mostrarán en una sola fila.

***Ejemplo: Obtener la marca y modelo (sin repetir) de todos los automóviles que fueron contratados y cuya fecha de finalización de contrato está dentro del año 2018.***

**SELECT marca, modelo FROM automoviles INNER JOIN contratos ON  
contratos.matricula = automoviles.matricula WHERE year(ffin)=2018 GROUP BY  
marca, modelo;**

marca	modelo
Mercedes	Clase C Coupe 170CV
Ford	Fiesta
Opel	Corsa
Seat	Leon
Audi	A4
Seat	Ibiza
Seat	Leon SW
Renault	Clio
BMW	318 TDI



## Agrupamiento de registros

*Ejemplo: Obtener las localidades en las que se tienen clientes.*

**SELECT localidad FROM clientes GROUP BY localidad;**

localidad
Cuenca
Madrid
Toledo

¿Qué utilidad podría tener esta consulta?

¿De qué forma alternativa podríamos obtener el mismo resultado?

## Agrupamiento de registros

*Ejemplo: Obtener el nombre y apellidos de los clientes que han realizado contratos a partir del 24 de diciembre de 2017. Los resultados deben estar ordenados ascendentemente por apellidos, nombre.*

**SELECT nombre, apellidos FROM clientes INNER JOIN contratos ON dnicliente=dni  
WHERE fin >='2017-12-24' GROUP BY **dnicliente** ORDER BY apellidos, nombre;**

nombre	apellidos
Soraya	Bats Corzo
Fanny	Cepeda
Eva	Coria García
Alicia	de la Hoz Gomez
Mariano	Dorado
Ana Belén	Fuentes Rojas
Beatriz	Garcia Martin
Carlos Javier	Lopez Carvajal
Ismael	Poza Rincón
Anais	Rodriguez
Vanessa	Rodriguez

¿Podríamos usar un **DISTINCT** en lugar del **GROUP BY**?

## Agrupamiento de registros

### Obtener cálculos sobre grupos de registros o filas

Cuando se realizan agrupamientos en una SELECT, podemos obtener cálculos sobre cada grupo con las funciones de resumen o agregado que hemos visto.

**Ejemplo: Obtener cuantos automóviles hay de cada marca usando la función count. Hay que agrupar por marca en una consulta sobre la tabla automoviles.**

**SELECT marca, count(\*) FROM automoviles GROUP BY marca;**

marca	count(*)
Audi	2
BMW	2
Citroen	1
Ford	2
Mercedes	3
Opel	1
Renault	2
Seat	4

## Agrupamiento de registros

***Ejemplo: Obtener el nombre y apellidos de los clientes que han realizado contratos a partir del 24 de diciembre de 2017 y cuantos contratos han realizado desde esa fecha. Los resultados deben estar ordenados ascendentemente por apellidos, nombre.***

**SELECT nombre, apellidos, count(\*) FROM clientes INNER JOIN contratos ON dnicliente=dni WHERE fini >='2016-12-27' GROUP BY dnicliente ORDER BY apellidos, nombre;**

nombre	apellidos	count(*)
Soraya	Bats Corzo	1
Fanny	Cepeda	1
Eva	Coria García	1
Alicia	de la Hoz Gomez	2
Mariano	Dorado	1
Ana Belén	Fuentes Rojas	1
Beatriz	Garcia Martin	1
Carlos Javier	Lopez Carvajal	2

## Agrupamiento de registros

*Ejemplo: Obtener el precio medio, precio máximo y precio mínimo de los coches de cada marca ordenados por precio medio descendentemente.*

**SELECT marca, avg(precio)AS medio, max(precio), min(precio) FROM automoviles  
GROUP BY marca ORDER BY medio DESC;**

marca	medio	max(precio)	min(precio)
Mercedes	152.43333180745444	167.87	123.65
Audi	121.47999954223633	124.2	118.76
BMW	119.11999893188477	121.79	116.45
Seat	89.22249794006348	102.63	70.56
Ford	78.13000106811523	87.62	68.64
Renault	73.13500022888184	92.65	53.62
Citroen	62.66999816894531	62.67	62.67
Opel	42.70000076293945	42.7	42.7

## Agrupamiento de registros

*Ejemplo: Obtener el precio medio, precio máximo y precio mínimo de los coches de la marca SEAT.*

```
SELECT avg(precio), max(precio), min(precio) FROM automoviles WHERE  
marca='SEAT';
```

avg(precio)	max(precio)	min(precio)
89.222500	102.63	70.56

### Poner condiciones sobre resultados de funciones de agrupamiento. Cláusula HAVING:

En una consulta se pueden seleccionar filas que cumplan condiciones relativas al resultado de una función de agrupamiento.

Detrás de **HAVING** se ha de escribir una condición de selección.

En la condición de selección sólo se pueden usar funciones de agrupamiento o resumen, columnas de agrupación (las que se utilicen con GROUP BY) o cualquier expresión basada en estas columnas o en las funciones de agrupamiento.

*Ejemplo: Obtener el número de clientes de cada localidad siempre que en la localidad haya más de tres clientes.*

```
SELECT localidad,count(*) FROM clientes GROUP BY localidad HAVING count(*)>3;
```

¿En qué cambia la sentencia si quitamos el HAVING?

## Agrupamiento de registros

*Ejemplo: Obtener las marcas de coches cuyo precio medio de alquiler sea inferior a 105 Euros.*

**SELECT marca FROM automoviles GROUP BY marca HAVING avg(precio)<105;**

marca
Ford
Seat



## Agrupamiento de registros

*Ejemplo: Obtener las marcas de coches y su precio medio de alquiler siempre que se cumpla que ese precio medio está comprendido entre 75 y 100 euros.*

**SELECT** marca,avg(precio) AS media **FROM** automoviles **GROUP BY** marca **HAVING** media >=75 **AND** media<=100;

marca	media
Ford	78.130000
Seat	89.222500

¿Cómo podemos escribir la condición del **HAVING** sin utilizar operadores de comparación?