

## **Actividad de aprendizaje 5-09**

### **Se trata de:**

- Realizar consultas SELECT que usan funciones MySQL numéricas o matemáticas y de cadenas de caracteres.

### **Objetivos:**

- Editar y ejecutar instrucciones SELECT que usan funciones MySQL.
- Utilizar funciones MySQL numéricas.
- Utilizar funciones MySQL de cadenas de caracteres.

### **Ejercicios**

Algunos de los ejercicios se resuelven sin necesidad de consultar tablas. Los ejercicios en los que se trabaja con tablas usan la base de datos **alquileres**.

1.- Obtener el resultado de truncar o redondear por abajo a entero el valor 4.7, redondearlo al entero más próximo y redondearlo a entero por arriba.

```
SELECT ceil(4.7) AS haciaArriba,  
floor(4.7) AS haciaAbajo,  
round(4.7) AS masProximo;
```

2.- Obtener la matrícula, marca, modelo de cada automóvil y el precio redondeado al entero más próximo.

```
SELECT matricula, marca, modelo, round(precio)  
FROM automoviles;
```

3.- Obtener el precio medio de alquiler de los automóviles de cada marca.

```
SELECT marca, avg(precio)  
FROM automoviles  
GROUP BY marca;
```

4.- Obtener el precio medio de alquiler de los automóviles de cada marca. Los precios medios deben salir con dos decimales redondeando al más próximo.

```
SELECT marca, round(avg(precio),2)  
FROM automoviles  
GROUP BY marca;
```

5.- Sabiendo que la duración en días de los contratos se obtiene con **datediff(ffin,fini)**, obtener sin decimales la duración media de los contratos finalizados.

```
SELECT floor(avg(datediff(ffin,fini))) AS duracionMedia  
FROM contratos WHERE ffin IS NOT NULL;
```

6.- Obtener un número aleatorio con 2 decimales comprendido entre 0 y 1 y un número aleatorio entero comprendido entre 1 y 50.

```
SELECT truncate(rand(),2), truncate(rand()*50+1,0);
```

7.- Obtener los datos de un contrato cuyo número de contrato sea igual a un número obtenido aleatoriamente inferior a 16.

```
SELECT *  
FROM contratos  
WHERE numcontrato=floor(rand()*15+1);
```

Esto no funciona porque, para cada contrato, se está generando un número aleatorio. Cuando el número que se genera coincide con el número de contrato que se está seleccionando, se pintan los datos del contrato. Si no coincide ninguno devuelve un resultado nulo.

En este caso se obtuvieron dos resultados (solo pedimos 1, pero al generarse un número aleatorio por cada contrato podemos obtener desde 0 coincidencias hasta n, siendo n el total de contratos).

	numcontrato	matricula	dnidiente	fini	ffin	kini	kfin
▶	1	1234JMY	08785691K	2017-12-12	2017-12-19	22200	22561
	9	2058JGF	09856064L	2017-12-27	2017-12-30	8150	9736
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

En este caso ninguno.

	numcontrato	matricula	dnidiente	fini	ffin	kini	kfin
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Cuando queramos obtener una fila aleatoriamente de una tabla o de varias filas de una tabla, basta con ordenar con por un valor random y limitar a cuantas filas queremos obtener aleatoriamente.

```
SELECT *  
FROM contratos  
WHERE numcontrato<16  
ORDER BY rand()  
LIMIT 1;
```

No obstante, de esta forma no se cumple la condición expresada en el enunciado, ya que no sabemos en realidad si existe una igualdad entre el número aleatorio y el de numcontrato.

La solución es crear y usar una variable:

```
SELECT @aleatorio:=floor(rand()*15+1) as aleatorio;  
SELECT * FROM contratos  
WHERE numcontrato = @aleatorio and numcontrato<16;
```

8.- Obtener los nombres y apellidos de los clientes que han realizado contratos. El nombre y apellidos debe salir en una sola columna con el formato APELLIDOS, NOMBRE y en mayúsculas. Para cada cliente se ha de obtener un número aleatorio entre 0 y 1.

```
SELECT ucase(concat(apellidos,', ',nombre)) AS nombreCompleto, rand() AS  
numeroAleatorio  
FROM clientes INNER JOIN contratos ON dni=dncliente;
```

9.- A partir de la consulta anterior, genera el resultado de un sorteo en el que resultan premiados tres clientes que hayan realizado contratos.

```
SELECT ucase(concat(apellidos,', ',nombre)) AS nombreCompleto, rand() AS  
numeroAleatorio  
FROM clientes INNER JOIN contratos ON dni=dncliente  
ORDER BY numeroAleatorio  
LIMIT 3;
```

10.- Obtener los nombres de clientes que tienen nombres compuestos, es decir, que tienen un espacio en la columna nombre (para separar los dos nombres).

```
SELECT nombre  
FROM clientes  
WHERE locate(' ',nombre)>0;
```

11.- Obtener nombres y apellidos de clientes que tienen en apellidos más de 16 caracteres.

```
SELECT nombre, apellidos  
FROM clientes  
WHERE char_length(apellidos)>12;
```

12.- Obtener los nombres y apellidos de todos los clientes. Del nombre solo se obtendrán los 8 primeros caracteres y de los apellidos los 12 primeros.

```
SELECT left(nombre,8), left(apellidos,12)  
FROM clientes;
```