

2 Consultas simples, funciones de agregado y expresiones regulares















10/01

*Vamos a mirar cosas sobre diferentes gestores: MariaDB y PostgreSQL

CONSULTAS SIMPLES

El operador punto (.) es para acceder a un campo:

- **SELECT c.customerName, c.contactLastName FROM customers c** (En este caso, estamos accediendo a los campos "customerName y contactLastName" de customers, es decir, devuelve el nombre y el primer apellido de todos los clientes. This statement selects the customerName and contactLastName from the customers table.)












				customerName	contactLastName
<input type="checkbox"/>	 Editor	 Copiar	 Borrar	Atelier graphique	Schmitt
<input type="checkbox"/>	 Editor	 Copiar	 Borrar	Signal Gift Stores	King
<input type="checkbox"/>	 Editor	 Copiar	 Borrar	Australian Collectors, Co.	Ferguson
<input type="checkbox"/>	 Editor	 Copiar	 Borrar	La Rochelle Gifts	Labruno
<input type="checkbox"/>	 Editor	 Copiar	 Borrar	Baane Mini Imports	Bergulfsen

- **SELECT c.customerName AS Nombre ,c.contactLastName AS Apellido1 FROM custommers c** (En este caso, estamos accediendo a los mismos campos que en la anterior consulta pero esta vez asignándole el nombre “Nombre” a “customerName” y el nombre “Apellido1” a “contactLastName”. The statement *SELECT c.customerName AS Nombre, c.contactLastName AS Apellido1 FROM custommers c* is asking to select the customer name and the contact last name from the table called "custommers c". The customer name will be referred to as "Nombre" and the contact last name will be referred to as "Apellido1".) **Ten cuidado con los espacios, si en vez de poner “Apellido1” pones “Apellido 1”, la consulta no funciona**


<div><div><div><div></div><div></div><div></div></div><div></div></div></div>							Nombre	Apellido1
<input type="checkbox"/>		Editar		Copiar		Borrar	Atelier graphique	Schmitt
<input type="checkbox"/>		Editar		Copiar		Borrar	Signal Gift Stores	King
<input type="checkbox"/>		Editar		Copiar		Borrar	Australian Collectors, Co.	Ferguson
<input type="checkbox"/>		Editar		Copiar		Borrar	La Rochelle Gifts	Labruno

- **SELECT p.productName AS NombreProducto, p.buyPrice AS precio FROM products p WHERE buyPrice<50** (En este caso, estamos haciendo lo mismo que con el anterior pero esta vez, le estamos añadiendo el WHERE que marca un filtro en donde el precio no puede ser mayor de 50, es decir, recupera nombre de producto y precio de los productos que valen menos de 50. NombreProducto and

precio are the names of two columns from the *products* table. The *WHERE* clause limits the results to products with a *buyPrice* less than 50.)

				NombreProducto	Precio
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1969 Harley Davidson Ultimate Chopper	48.81
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1970 Plymouth Hemi Cuda	31.92
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1936 Mercedes-Benz 500K Special Roadster	24.26
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1911 Ford Town Car	33.30

- **SELECT** **p.productName AS NombreProducto**, **p.buyPrice AS Precio** **FROM** **products p** **WHERE buyPrice >=40 AND buyPrice <=50** (En este caso, estamos haciendo lo mismo que con el anterior pero esta vez, recupera nombre de producto y precio de los productos que valen entre 40 y 50, ya que usamos el operador **AND** como conector. This statement is asking for the product name and price of all products that have a price between \$40 and \$50.)

<div><div><div></div><div></div><div></div></div></div>				NombreProducto	Precio			
<input type="checkbox"/>		Editar		Copiar		Borrar	1969 Harley Davidson Ultimate Chopper	48.81
<input type="checkbox"/>		Editar		Copiar		Borrar	P-51-D Mustang	49.00
<input type="checkbox"/>		Editar		Copiar		Borrar	1969 Dodge Super Bee	49.05
<input type="checkbox"/>		Editar		Copiar		Borrar	1932 Alfa Romeo 8C2300 Spider Sport	43.26
<input type="checkbox"/>		Editar		Copiar		Borrar	1970 Chevy Chevelle SS 454	49.24

- **SELECT** **p.productName AS NombreProducto**, **p.productLine AS Linea**, **p.buyPrice AS Precio** **FROM** **products p** **WHERE buyPrice BETWEEN 40 AND 50** (En este caso, estamos haciendo lo mismo que con el anterior pero esta vez, en vez de usar **AND** como conector, usamos **BETWEEN** y añadimos un parámetro más que en este caso sería el *Line*. The statement is selecting the product name, product line, and buy price from the *products* table, where the buy price is between

40 and 50.) **Ten cuidado con las comas, si las enumeraciones van sin comas, luego no se registra bien y da error**

<div><div><div></div><div></div><div></div></div></div>				NombreProducto	Line	Precio			
<div><div><div></div></div></div>	<div><div><div></div></div></div>	Editar	<div><div><div></div></div></div>	Copiar	<div><div><div></div></div></div>	Borrar	1969 Harley Davidson Ultimate Chopper	Motorcycles	48.81
<div><div><div></div></div></div>	<div><div><div></div></div></div>	Editar	<div><div><div></div></div></div>	Copiar	<div><div><div></div></div></div>	Borrar	P-51-D Mustang	Planes	49.00
<div><div><div></div></div></div>	<div><div><div></div></div></div>	Editar	<div><div><div></div></div></div>	Copiar	<div><div><div></div></div></div>	Borrar	1969 Dodge Super Bee	Classic Cars	49.05
<div><div><div></div></div></div>	<div><div><div></div></div></div>	Editar	<div><div><div></div></div></div>	Copiar	<div><div><div></div></div></div>	Borrar	1932 Alfa Romeo 8C2300 Spider Sport	Vintage Cars	43.26

- **SELECT** p.productName AS NombreProducto, p.buyPrice AS Precio **FROM** products p **WHERE** (buyPrice>=40 AND buyPrice<=50) OR buyPrice=89.14 (En este caso, estamos haciendo lo mismo que con el anterior pero esta vez, usando el operador OR, con el que tenemos que tener cuidado con los paréntesis, y añadimos también los productos que tienen un valor exacto a 89.14. This statement retrieves the product name and price of products with a price between 40 and 50, or with a price of 89.14.)



















<div><div><div>←</div><div>T</div><div>→</div></div></div>				NombreProducto	Precio
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1969 Harley Davidson Ultimate Chopper	48.81
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1969 Corvair Monza	89.14
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	P-51-D Mustang	49.00
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1969 Dodge Super Bee	49.05
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1932 Alfa Romeo 8C2300 Spider Sport	43.26

- **SELECT** p.productName AS NombreProducto, p.productLine AS Linea, p.buyPrice AS Precio **FROM** products p **WHERE** p.productLine LIKE "Vin%ge Cars"(En este caso, estamos buscando productos que coincidan con el patrón de búsqueda dado, utilizando el comodín %. This statement will find the product name,

product line, and buy price of all products whose product line starts with "Vin" and ends with "ge Cars".)

<div><div><div>←T→</div></div></div>				NombreProducto	Line	Precio
<input type="checkbox"/>	<div><div><div></div></div>Editar</div>	<div><div><div></div></div>Copiar</div>	<div><div><div></div></div>Borrar</div>	1937 Lincoln Berline	Vintage Cars	60.62
<input type="checkbox"/>	<div><div><div></div></div>Editar</div>	<div><div><div></div></div>Copiar</div>	<div><div><div></div></div>Borrar</div>	1936 Mercedes-Benz 500K Special Roadster	Vintage Cars	24.26
<input type="checkbox"/>	<div><div><div></div></div>Editar</div>	<div><div><div></div></div>Copiar</div>	<div><div><div></div></div>Borrar</div>	1917 Grand Touring Sedan	Vintage Cars	86.70
<input type="checkbox"/>	<div><div><div></div></div>Editar</div>	<div><div><div></div></div>Copiar</div>	<div><div><div></div></div>Borrar</div>	1911 Ford Town Car	Vintage Cars	33.30
<input type="checkbox"/>	<div><div><div></div></div>Editar</div>	<div><div><div></div></div>Copiar</div>	<div><div><div></div></div>Borrar</div>	1932 Model A Ford J-Coupe	Vintage Cars	58.48

- **SELECT** p.productName **AS** NombreProducto, p.productLine **AS** Linea, p.buyPrice **AS** Precio **FROM** products p **WHERE** p.productLine **LIKE** "Vin%ge_ars"(Cuando ponemos % significa que puede ir una letra o varias en el sitio, pero cuando ponemos el guion bajo "_" significa que sustituye únicamente a una letra. When we put % it means that one or several letters can go in that place, but when we put the underscore "_" it means that it only substitutes one letter.)

<div><div><div></div><div></div><div></div></div></div>				NombreProducto	Line	Precio
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1937 Lincoln Berline	Vintage Cars	60.62
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1936 Mercedes-Benz 500K Special Roadster	Vintage Cars	24.26
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1917 Grand Touring Sedan	Vintage Cars	86.70
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1911 Ford Town Car	Vintage Cars	33.30
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1932 Model A Ford J-Coupe	Vintage Cars	58.48
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1928 Mercedes-Benz SSK	Vintage Cars	72.56

CONSULTAS CON FUNCIONES DE AGREGADO COUNT, SUM Y AVG

- **SELECT COUNT(e.employeeNumber) FROM employee** (Cuenta a los empleados según su primary key por lo que estarán todos contados. This statement counts the number of employees in the employee table.) **Cuidado con los nulos porque no los cuenta**

COUNT(employeeNumber)
23

- **SELECT COUNT(e.employeeNumber) FROM employee WHERE firstName LIKE "G%"** (Cuenta los empleados cuyo nombre empieza por G. This statement counts the number of employees whose first name begins with the letter "G".)

COUNT(employeeNumber)
3

- **SELECT SUM(p.amount) FROM payments** (Suma todos los pagos. The command **SELECT SUM(p.amount) FROM payments** will calculate the total amount of all payments in the payments table.)

SUM(amount)
8853839.23

- **SELECT SUM(p.amount) FROM payments p WHERE p.customerNumber=103** (En este caso solo sumaríamos los pagos del cliente con número 103. The statement is asking to find the total amount of payments made by customer number 103.)

SUM(amount)

22314.36

- **SELECT AVG(p.buyPrice) FROM products p** (Media del precio de nuestro productos. The average buy price of all products in the products table is being calculated.)

AVG(p.buyPrice)

54.395182

- **SELECT AVG(p.buyPrice) FROM products p WHERE p.productLine = "Vintage Cars"** (Media del precio de los productos de la linea "Vintage Cars". The average buy price of all products in the "Vintage Cars" product line will be calculated.)

AVG(p.buyPrice)

46.066250

- **SELECT AVG(p.buyPrice) FROM products p WHERE p.productLine = "Vintage Cars" AND p.buyPrice < 50** (Media del precio de los productos de la linea "Vintage Cars" con un precio inferior a 50. This statement calculates the average buy price of products from the "Vintage Cars" product line that have a buy price less than 50€.)

AVG(p.buyPrice)

31.166923

- **SELECT AVG(p.buyPrice) FROM products p WHERE (p.buyPrice BETWEEN 20 AND 40) OR (p.buyPrice BETWEEN 60 AND 100)** (Media del precio de los productos en los intervalos de 20 a 40 euros y 60 a 100. The average of the buy

prices of products that are either between 20 and 40 or between 60 and 100 is being calculated.)










AVG(p.buyPrice)
54.986184

CONSULTAS CON EXPRESIONES REGULARES (REGEXP)

- **SELECT * FROM** `employees e` **WHERE e.firstName REGEXP** **"Jeff|Mary|Diane"**
(Muestra los empleados que se llamen Jeff Mary o Diane. This statement will select all information from the table called "employees" for people whose first name is either Jeff, Mary, or Diane.)

<div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div></div></div>				employeeNumber	lastName	firstName	extension	email
<div><div><div></div></div></div>	<div><div></div><div>Editar</div></div>	<div><div></div><div>Copiar</div></div>	<div><div></div><div>Borrar</div></div>	1002	Murphy	Diane	x5800	dmurph
<div><div><div></div></div></div>	<div><div></div><div>Editar</div></div>	<div><div></div><div>Copiar</div></div>	<div><div></div><div>Borrar</div></div>	1056	Patterson	Mary	x4611	mpatt
<div><div><div></div></div></div>	<div><div></div><div>Editar</div></div>	<div><div></div><div>Copiar</div></div>	<div><div></div><div>Borrar</div></div>	1076	Firrelli	Jeff	x9273	jfirrelli

- **SELECT * FROM** `employees e` **WHERE e.firstName REGEXP** **"^Ge"** (Muestra los empleados cuyo nombre empieza por Ge. This statement is used to find all employees whose first name starts with "Ge".)

	employeeNumber	lastName	firstName	extension	email
<input type="checkbox"/>  Editar  Copiar  Borrar	1102	Bondur	Gerard	x5408	gbondur@classicmode
<input type="checkbox"/>  Editar  Copiar  Borrar	1323	Vanauf	George	x4102	gvanauf@classicmodel
<input type="checkbox"/>  Editar  Copiar  Borrar	1370	Hernandez	Gerard	x2028	ghernande@classicmo

- **SELECT * FROM** `employees e` **WHERE** `e.firstName REGEXP "ff$"` (Muestra los empleados cuyo nombre acabe en ff. This statement is used to find all employees whose first name ends with "ff".)

	employeeNumber	lastName	firstName	extension	email
<input type="checkbox"/>  Editar  Copiar  Borrar	1076	Firrelli	Jeff	x9273	jfirrelli@classicmodelc







- **SELECT * FROM** `employees e` **WHERE** `e.firstName REGEXP "Jef*"` (Mostraría todos aquellos empleados cuyo nombre empiece por "Je" y que tengan 0 o N efes. This statement retrieves all records from the 'employees' table where the first name starts with "Jef".)

	employeeNumber	lastName	firstName	extension	email
<input type="checkbox"/>  Editar  Copiar  Borrar	1076	Firrelli	Jeff	x9273	jfirrelli@classicmodelc




- **SELECT * FROM** `employees e` **WHERE** `e.firstName REGEXP "(lia)+"` (Mostraría los empleados cuyo nombre contenga la secuencia de caracteres "lia" al menos un vez. This statement will select all columns from the table called "employees" and look for any first names that contain the letters "lia" in any order.)

←T→	employeeNumber	lastName	firstName	extension	email
<input type="checkbox"/>  Editar  Copiar  Borrar	1088	Patterson	William	x4871	wpatterson@classicmo

- **SELECT * FROM products p WHERE p.productName REGEXP '^([0-9]{2}[A-Z]{1})'** ([0-9] Mostraría los productos cuyo nombre empieza por 2 números seguidos de una letra. El operador ^ nos indica que tiene que ser al principio todo el patrón entre paréntesis, luego [0-9] admitiría cualquier dígito y al llevar a continuación {2} eso obliga a que aparezca 2 veces, por último obligamos con [A-Z]{1} a que a continuación de los 2 número venga una letra cualquiera. This statement will select all columns from the 'products' table where the product name starts with two numbers followed by one letter.). **Si ponemos en vez de [0-9] sin el guion [09] en vez de ser un intervalo, es un número que o es 0 o es 9. Si ponemos [abc] solo valen las letras “a”, “b” o “c”, las otras no. Para poner un paréntesis, coma o punto se puede poner así: [, .] e incluso repetirlo varias veces: [, .]{2}**

←T→	productCode	productName	productLine	productScale	productVendor
<input type="checkbox"/>  Editar  Copiar  Borrar	S18_3136	18th Century Vintage Horse Carriage	Vintage Cars	1:18	Red Start Diecas
<input type="checkbox"/>  Editar  Copiar  Borrar	S24_2011	18th century schooner	Ships	1:24	Carousel DieCas Legends

- **SELECT * FROM products p WHERE p.productName REGEXP '([0-9]{4})\$'** (Recupera los productos cuyo nombre tenga 4 dígitos al final de la expresión. This statement retrieves all the products from the "products" table whose product name ends with a 4-digit number.)

←T→	productCode	productName	productLine	productScale	productVendor
<input type="checkbox"/>  Editar  Copiar  Borrar	S10_1949	1952 Alpine Renault 1300	Classic Cars	1:10	Classic Metal Creations

