

# 1 Consultas sencillas SQL

*\*En Php my Admin, en Importar, tenemos que ir a Otras opciones y deshabilitar la revisión de las claves foráneas con el scrip original, con el scrip del aula virtual está bien*

Cosas básicas que saber de SQL:

- Existen muchos tipos de SQL
- Se trató de estandarizar, el último es del 2016
- Es un lenguaje de consultas y búsquedas
- Se pueden hacer búsquedas regulares por cadenas complejas
- Tienen características XML y JSON
- Permite consultas recursivas
- Vamos a mirar sobre MySQL y PostgreSQL
- El Cliente SQL es un scrip del servidor, no es el cliente per se

Podemos encontrarnos cosas que funcionan en diferentes servidores, pero que no son compatibles con otros, lo mejor es usar el SQL estándar

Vamos a empezar no por el lenguaje definición, sino con el lenguaje de consulta. Por lo que vamos a empezar haciendo consultas y luego continuaremos revisando cómo crear

tablas y bases de datos con SQL


Necesitamos saber el lenguaje de manipulación de datos y el de consulta que son las **SELECT** y hay que dominarlas bien. Empezamos con las más sencillas que solo influyen en una tabla con algunas restricciones.

Veremos también bases no SQL con MongoDB

De MariaDB, hay un manual de SQL en la página principal:

#### Documentación MariaDB Server

Content reproduced on this site is the property of its respective owners, and this content is not reviewed in advance by MariaDB. The views, information and opinions expressed by this content do not necessarily represent those of MariaDB or any other party.

 <https://mariadb.com/kb/es/documentacion-mariadb-server/>

#### Regular Expressions Overview

Regular Expressions allow MariaDB to perform complex pattern matching on a string.

 <https://mariadb.com/kb/en/regular-expressions-overview/>

Una consulta plana es solo sobre una tabla, le dices lo que quieres ver con el **SELECT** (con **\*** es todo o campo a campo).

\*Mirar alias de las tablas y la teoría cartesiana

\*Mirar **REGEXP** y las regular expressions

\*Hemos mirado varios comandos como: **AND, OR, LIKE, BETWEEN...**

En PHPMyAdmin, si hacemos click en una tabla nos aparece esto:

```
SELECT * FROM customers WHERE 1
```







El **WHERE** es **1** porque es una coetilla y el asterisco sustituye a todos los campos

---

## Consultas de 21-22/12/2022

- **SELECT \* FROM customers WHERE contactFirstName='Jean'** (Busca gente que se llame Jean. This statement will select all information from the "customers" table)













where the contact first name is "Jean".)

	customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	address
<input type="checkbox"/>   	112	Signal Gift Stores	King	Jean	7025551838	8489 Strong St.	NULL
<input type="checkbox"/>   	233	Québec Home Shopping Network	Fresnière	Jean	(514) 555-8054	43 rue St. Laurent	NULL

- **SELECT \* FROM** `customers` **WHERE** `contactFirstName LIKE 'J%'` (Así empieza por J. This statement will select all the columns from the customers table where the contact first name starts with the letter "J".)

	customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCo
<input type="checkbox"/>   	112	Signal Gift Stores	King	Jean	7025551838	8489 Strong St.	NULL	Las Vegas	NV	83030
<input type="checkbox"/>   	119	La Rochelle Gifts	Labruno	Janine	40.67.8555	67, rue des Cinquante Otages	NULL	Nantes	NULL	44000
<input type="checkbox"/>   	121	Baane Mini Imports	Bergulfsen	Jonas	07-98 9555	Erling Skakkes gate 78	NULL	Stavern	NULL	4110
<input type="checkbox"/>   	129	Mini Wheels Co.	Murphy	Julie	6505555787	5557 North Pendale Street	NULL	San Francisco	CA	94217
<input type="checkbox"/>   	145	Danish Wholesale Imports	Petersen	Jytte	31 12 3555	Vinbæltet 34	NULL	Kobenhavn	NULL	1734
<input type="checkbox"/>   	151	Muscle Machine Inc	Young	Jeff	2125557413	4092 Furth Circle	Suite 400	NYC	NY	10022
<input type="checkbox"/>   	161	Technics Stores Inc.	Hashimoto	Juri	6505556809	9408 Furth Circle	NULL	Burlingame	CA	94217
<input type="checkbox"/>   	173	Cambridge Collectables Co.	Tseng	Jerry	6175555555	4658 Baden Av.	NULL	Cambridge	MA	51247
<input type="checkbox"/>   	175	Gift Depot Inc.	King	Julie	2035552570	25593 South Bay Ln.	NULL	Bridgewater	CT	97562
<input type="checkbox"/>   	205	Toys4GrownUps.com	Young	Julie	6265557265	78934 Hillside Dr.	NULL	Pasadena	CA	90003
<input type="checkbox"/>   	233	Québec Home Shopping Network	Fresnière	Jean	(514) 555-8054	43 rue St. Laurent	NULL	Montréal	Québec	H1J 1C3
<input type="checkbox"/>   	299	Norway Gifts By Mail, Co.	Klaeboe	Jan	+47 2212 1555	Drammensveien 126A	PB 211 Sentrum	Oslo	NULL	N 0106
<input type="checkbox"/>   	321	Corporate Gift Ideas Co.	Brown	Julie	6505551386	7734 Strong St.	NULL	San Francisco	CA	94217
<input type="checkbox"/>   	344	CAF Imports	Fernandez	Jesus	+34 913 728 555	Merchants House	27-30 Merchant's Quay	Madrid	NULL	28023









































- **SELECT \* FROM** `customers` **WHERE** `contactFirstName LIKE 'J%E'` (Empieza por J y acaba por E. This statement will select all columns from the customers table where the contact first name starts with "J" and ends with "E".)

	customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLi
<input type="checkbox"/>  Editar  Copiar  Borrar	129	Mini Wheels Co.	Murphy	Julie	6505555787	5557 North Pendale Street	NULL
<input type="checkbox"/>  Editar  Copiar  Borrar	175	Gift Depot Inc.	King	Julie	2035552570	25593 South Bay Ln.	NULL
<input type="checkbox"/>  Editar  Copiar  Borrar	205	Toys4GrownUps.com	Young	Julie	6265557265	78934 Hillside Dr.	NULL
<input type="checkbox"/>  Editar  Copiar  Borrar	321	Corporate Gift Ideas Co.	Brown	Julie	6505551386	7734 Strong St.	NULL


























- **SELECT** customerName, contactFirstName, contactLastName **FROM** customers **WHERE** contactFirstName LIKE 'J%E' (Aquí seleccionamos de los tres puntos que nos interesan, por eso solo vemos las tres tablas. This statement retrieves the customer name, first name, and last name of customers whose first name starts with "J" and ends with "E".)

	customerName	contactFirstName	contactLastName
<input type="checkbox"/>  Editar  Copiar  Borrar	Mini Wheels Co.	Julie	Murphy
<input type="checkbox"/>  Editar  Copiar  Borrar	Gift Depot Inc.	Julie	King
<input type="checkbox"/>  Editar  Copiar  Borrar	Toys4GrownUps.com	Julie	Young
<input type="checkbox"/>  Editar  Copiar  Borrar	Corporate Gift Ideas Co.	Julie	Brown

- **SELECT \* FROM** orderdetails **WHERE** priceEach < 50 (Pone los precios menores de 50. This statement will select all data from the orderdetails table where the price of each item is less than 50.)

		orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber
<input type="checkbox"/>	 Editar  Copiar  Borrar	10100	S24_3969	49	35.29	1
<input type="checkbox"/>	 Editar  Copiar  Borrar	10101	S24_1937	45	32.53	3
<input type="checkbox"/>	 Editar  Copiar  Borrar	10101	S24_2022	46	44.35	2
<input type="checkbox"/>	 Editar  Copiar  Borrar	10102	S18_1367	41	43.13	1
<input type="checkbox"/>	 Editar  Copiar  Borrar	10103	S18_4668	41	40.75	9
<input type="checkbox"/>	 Editar  Copiar  Borrar	10104	S24_2840	44	30.41	10
<input type="checkbox"/>	 Editar  Copiar  Borrar	10105	S72_3212	25	44.77	8
<input type="checkbox"/>	 Editar  Copiar  Borrar	10106	S50_1341	39	35.78	6
<input type="checkbox"/>	 Editar  Copiar  Borrar	10106	S72_1253	48	43.70	15
<input type="checkbox"/>	 Editar  Copiar  Borrar	10108	S32_2206	27	36.21	13
<input type="checkbox"/>	 Editar  Copiar  Borrar	10109	S24_2972	29	32.10	6
<input type="checkbox"/>	 Editar  Copiar  Borrar	10110	S24_1628	29	43.27	15
<input type="checkbox"/>	 Editar  Copiar  Borrar	10110	S24_1937	20	28.88	3

- **SELECT \* FROM** `orderdetails` **WHERE priceEach BETWEEN 50.7 AND 65** (Pone los precios entre los valores que se dan. This statement is used to retrieve information from a database. It will return all the data from the 'orderdetails' table where the price of each item is between 50.7 and 65.)

		orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber
<input type="checkbox"/>	 Editar  Copiar  Borrar	10100	S18_2248	50	55.09	2
<input type="checkbox"/>	 Editar  Copiar  Borrar	10103	S18_2432	22	58.34	2
<input type="checkbox"/>	 Editar  Copiar  Borrar	10103	S18_2957	35	61.84	14
<input type="checkbox"/>	 Editar  Copiar  Borrar	10103	S32_3522	45	63.35	7
<input type="checkbox"/>	 Editar  Copiar  Borrar	10104	S24_1444	35	52.02	6
<input type="checkbox"/>	 Editar  Copiar  Borrar	10104	S32_2509	35	51.95	11
<input type="checkbox"/>	 Editar  Copiar  Borrar	10104	S32_3207	49	56.55	4
<input type="checkbox"/>	 Editar  Copiar  Borrar	10104	S50_1514	32	53.31	2

- **SELECT COUNT(\*) FROM** `orderdetails` **WHERE priceEach BETWEEN 50.7 AND 65** (Es un operador que cuenta, es un operador de agregado que cuenta todo en donde no haya nulos. Se puede poner la clave primaria siempre porque así nunca hay nulos. Contamos con asteriscos o con clave primaria para evitar problemas. This statement counts the number of entries in the orderdetails table where the price of each item is between 50.7 and 65.)

```
COUNT(*)
467
```

- **USE+ base de datos** (Así abrimos y seleccionamos la base de datos)

```
1 USE customers
```

- **SELECT c.\*, o.addressLine1 FROM** `customers c` , `offices o` (Así asignamos los alias [`customers = c` y `offices = o`] y elegimos la info que queremos extraer de cada base de datos. Aquí se hace así porque no hay ninguna condición. This statement selects all columns (\*) from the table customers c and the addressLine1 column from the table offices o.)

customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmployeeNumber	creditLimit	addressLine1
103	Atelier graphique	Schmitt	Carine	40 32 2555	54, rue Royale	NULL	Nantes	NULL	44000	France	1370	21000.00	100 Market Street
103	Atelier graphique	Schmitt	Carine	40 32 2555	54, rue Royale	NULL	Nantes	NULL	44000	France	1370	21000.00	1550 Court Place
103	Atelier graphique	Schmitt	Carine	40 32 2555	54, rue Royale	NULL	Nantes	NULL	44000	France	1370	21000.00	523 East 53rd Street
103	Atelier graphique	Schmitt	Carine	40 32 2555	54, rue Royale	NULL	Nantes	NULL	44000	France	1370	21000.00	43 Rue Jouffroy D'abbe

- **SELECT e.\*, o.addressLine1 FROM** `employees e` , `offices o` **WHERE e.officeCode=o.officeCode** (Así hacemos que busquen de ambas bases de datos la info en donde coincida el código de oficina de empleados con el de las oficinas, y nos muestre las tablas seleccionadas. The statement is asking to select all columns from the table "employees" (e.\*) and the column "addressLine1" from the table

"offices" (o.addressLine1) where the "officeCode" from the "employees" table (e.officeCode) matches the "officeCode" from the "offices" table (o.officeCode).)

employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	addressLine1
1002	Murphy	Diane	x5800	dmurphy@classicmodelcars.com	1	NULL	President	100 Market Street
1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1002	VP Sales	100 Market Street
1076	Firrelli	Jeff	x9273	jfirrelli@classicmodelcars.com	1	1002	VP Marketing	100 Market Street
1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	100 Market Street
1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1143	Sales Rep	100 Market Street
1166	Thompson	Leslie	x4065	lthompson@classicmodelcars.com	1	1143	Sales Rep	100 Market Street
1188	Firrelli	Julie	x2173	jfirrelli@classicmodelcars.com	2	1143	Sales Rep	1550 Court Place

- **SELECT** employees.lastName, employees.firstName **FROM** employees **WHERE** employees.employeeNumber=1588 (Así nos enseña el nombre y apellido de los empleados y añadimos una restricción para que solo muestren los empleados con el número de empleado 1588. Get the last name and first name of the employee with employee number 1588 from the employees table.)

























También podemos poner la consulta de SQL así con comillas simples en el número. Aunque lo estuviéramos tratando como un char, PHPMyAdmin hace una conversión interna de tipos de variables, por lo que no habría problema

- **SELECT** employees.lastName, employees.firstName **FROM** employees **WHERE** employees.employeeNumber='1588'

←T→	▼	lastName	firstName
<input type="checkbox"/>	Editar	Copiar	Borrar
	Patterson	William	

- **SELECT** employees.lastName, employees.firstName **FROM** employees **WHERE** employees.employeeNumber>1000 **AND** employees.employeeNumber<1500 (Así buscamos todos los empleados que tengan un número de empleado entre 1000 y 1500. This statement retrieves the last name and first name of employees whose employee number is between 1000 and 1500.)



<div><div><div><div><div></div><div></div></div><div></div></div></div><div>lastName</div><div>firstNam</div></div>					
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Murphy	Diane
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Patterson	Mary
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Firrelli	Jeff
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Patterson	William
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Bondur	Gerard
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Bow	Anthony
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Jennings	Leslie
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	Thompson	Leslie

- **SELECT** employees.lastName, employees.firstName FROM employees WHERE employees.employeeNumber>1000 AND employees.employeeNumber<1500 AND employees.firstName LIKE 'W%' (Así busca como la anterior pero solo los nombres que empiecen por W. This statement will select the last name and first name of employees whose employee number is between 1000 and 1500 and whose first name starts with the letter "W".)

		employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle
<input type="checkbox"/>	 Editor  Copiar  Borrar	1088	Patterson	William	x4871	wpatterson@classicmodelcars.com	6	1056	Sales Manager (APAC)

- **SELECT** employees.lastName, employees.firstName FROM employees WHERE employees.employeeNumber>1000 AND employees.employeeNumber<1500 AND (employees.firstName LIKE 'W%' OR employees.firstName LIKE 'M%') (Así busca como la anterior pero solo los nombres que empiecen por W o M, y con esto podemos aprender a usar también los paréntesis y la preferencia de operadores. This statement will select the last name and first name of employees whose employee number is between 1000 and 1500 and whose first name starts with either W or M.)



	employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1002	VP Sales
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1088	Patterson	William	x4871	wpatterson@classicmodelcars.com	6	1056	Sales Manager (APAC)

- **SELECT COUNT(employees.employeeNumber) FROM employees** (Así cuenta los empleados que existen en la empresa con el atributo clave de la base de datos employees. This statement counts the number of employees in the "employees" table.)

```
COUNT(employees.employeeNumber)
```

23

- **SELECT COUNT(employees.reportsTo) FROM employees** (Así cuenta los empleados según si tienen jefe o no, un atributo que no es clave de la base de datos, por lo que no se hace un recuento de todos los empleados ya que el resultado es 22 y no 23. El atributo reportsTo, tiene un nulo, cosa que el atributo clave, nunca va a tener. This statement counts the number of employees who have a manager assigned to them.)

```
COUNT(employees.reportsTo)
```











22

- **SELECT COUNT(DISTINCT employees.firstName) FROM employees** (Con el operador DISTINCT, no se cuentan los atributos que sean iguales, en este caso, que tengan el mismo nombre, por lo que va a contar todos los nombres excepto los repetidos. En este caso vemos que hay dos personas con el mismo nombre. This statement counts the number of unique first names in the employees table.)

```
COUNT(DISTINCT employees.firstName)
```

21

- **SELECT DISTINCT employees.firstName FROM employees** (Aquí podemos ver cómo el operador **DISTINCT** se puede usar sin el **COUNT**, para enseñar todos los nombres, sin repetidos, de los empleados. This statement will select all the unique first names of employees from the employees table.)

	firstName
<input type="checkbox"/>  Editar  Copiar  Borrar	Diane
<input type="checkbox"/>  Editar  Copiar  Borrar	Mary
<input type="checkbox"/>  Editar  Copiar  Borrar	Jeff
<input type="checkbox"/>  Editar  Copiar  Borrar	William
<input type="checkbox"/>  Editar  Copiar  Borrar	Gerard

- **SELECT AVG(products.buyPrice) FROM products** (AVG es el operador para hacer medias, con este, podemos ver la media del coste de los productos. The average of the buyPrice of all products in the products table is being calculated.)

AVG(products.buyPrice)
54.395182

- **SELECT AVG(products.buyPrice) FROM products WHERE products.productLine='Classic cars'** (Así hacemos la media del precio únicamente de los coches clásicos. This statement calculates the average buy price of all products in the "Classic cars" product line.)

AVG(products.buyPrice)
64.446316

- **SELECT AVG(products.buyPrice) FROM products WHERE products.productLine='Classic cars' OR products.productLine='Planes'**  
(Hacemos la media de los precios de los coches clásicos y de los aviones. This statement will calculate the average buy price of products that are either Classic cars or Planes.)

También podemos hacer la misma consulta con un alias de la siguiente manera:

- **SELECT AVG(p.buyPrice) FROM products p WHERE p.productLine='Classic cars' OR p.productLine='Planes'** (Tenemos que tener cuidado con dónde poner la abreviatura ya que va en todos los casos excepto en donde se introduce que es en el operador FROM)

AVG(products.buyPrice)
60.890200

- **SELECT AVG(p.buyPrice) AS media FROM products p WHERE p.productLine='Classic cars' OR p.productLine='Planes'** (Podemos también darle alias a los operadores con AS, para que sean más fáciles de interpretar los datos. This statement calculates the average (AVG) buyPrice of products (p) where the productLine is either Classic cars or Planes. The result is stored in the variable media.)

media
60.890200