

Con esta clase vamos a terminar nuestro módulo de bases de datos.

Ahora, vamos a importarnos una base de datos y sus correspondientes tablas de muestra que tienen mucha información, muchísimo más que la que hemos creado nosotros a manera de ejemplo y la vamos a utilizar para hacer consultas más elaboradas. De eso se trata de esta clase, es una clase eminentemente práctica, con consultas, consultas SQL o "Queries" como se les llama en el ambiente de trabajo de las bases de datos no relacionales. Así que, ¡a practicar!

Perfecto. Para hacer consultas una serie de datos, usos de la instrucción "select" necesitamos una tabla que tenga o, mejor dicho, una base de datos que tenga una o unas cuantas tablas y muchos registros. Sería muy tedioso crear esto, por lo tanto vamos a importar una de las bases de datos de muestra que vienen con estos administradores de bases de datos. Tenemos una que se llama "northwind", o viento norte, no le hemos puesto nosotros el nombre, es una base de datos bastante conocida, por otra parte, que tiene muchas tablas con unos cuantos registros.

Para importar una base de datos, lo primero que tenemos que hacer aquí es crear una nueva base de datos o podríamos crearla desde la línea de comandos "create data base" y aquí pongámosle el nombre "northwind", okey, que es el nombre original de esta base de datos. Por otra parte, bien, la creamos y aquí ya nos aparece, perfecto, seleccionamos esta base de datos. Venimos aquí, hacemos click, vemos que está seleccionada, base de datos "northwind" y vamos a importar y, a continuación, seleccionar archivo. Aquí dice "buscar en su ordenador", tenemos una copia de esta base de datos por aquí, en nuestra carpeta de "web development" en MySQL y aquí está la base de datos "northwind", con extensión, punto. Aquí pueden ver la extensión, es "punto SQL", okey, vamos a darle click aquí, "abrir" y simplemente "continuar". Se va a producir la importación de nuestra base de datos y quedará guardada aquí en "northwind". Bueno, puede demorar un poco, listo, porque es una base de datos bastante grande. Pueden ver aquí 3.219 consultas ejecutadas. Okey, y aquí tenemos "northwind", miren, tiene muchas tablas, esto nos va a venir muy bien para, bueno, para practicar, para hacer algunas consultas aquí. Hasta ahora hemos visto la forma más sencilla de "select", es decir, "select" junto con el modificador "asterisco", que significa "seleccioname todo" de alguna tabla particular. Aquí podemos trabajar con cualquiera de estas, por ejemplo, "proveedores", se llama "suppliers" y esto nos va a mostrar todos los registros de esa tabla, que son muchos, como pueden ver. Perfecto, vamos a ver formas un poquito más complejas. Podríamos indicar que no queremos ver todas las columnas, a lo mejor queremos ver solamente la columna "company name" y "contact name" o "company name" y "city" y por ejemplo, bien, podemos hacerlo podemos indicar "company name", "company name", una coma y la ciudad "city" y luego sí, otra vez de la tabla "suppliers", "company name" y "city". Bien, noten que aquí tenemos todos los valores, todos los registros de esa tabla, pero solamente nos muestra las columnas especificadas por nosotros. Bueno, podemos hacer muchas más cosas, por supuesto.

Vamos a ver, podríamos seleccionar, podríamos utilizar el método "select distinct", para que ustedes lo vean, se usa para retornar solamente los valores que son distintos o diferentes, en el caso de que haya duplicados. Por ejemplo, podríamos buscar aquí en nuestras tablas, la tabla "customers", "customers" y buscar si hay algún campo candidato, miren aquí, donde

dice, por ejemplo, "country", tenemos el valor de, desde el país, repetido, aquí hay un comprador de Alemania, otro y otro, se repiten, okey, podríamos hacer lo siguiente, sólo para probar esta instrucción, ¿okey?, esto lo voy a cerrar para tener más espacio. Aquí podríamos indicar, volviendo a nuestra línea de consultas, bien, esto puedo hacerlo más pequeño también, ¿no?, podríamos indicar aquí lo siguiente, déjenme, pensar hagamos un "select distinct", bueno, podríamos indicar todas las columnas o algunas nada más, vamos a dejar que nos muestre todas las columnas. Aquí, si buscásemos todos los registros por el campo "país", obtendríamos lo siguiente, vamos a ver, primero hagámoslo así, "select country from costumers" ¿que obtendríamos en este caso?, bueno, vemos como solamente pedimos el campo país, vemos ese campo de todos los registros y aquí podemos ver claramente que hay países que se repiten, lo cual es perfectamente lógico, ¿okey? No hay ningún problema con ello, solamente quiero hacerles ver dónde está esa información, para que entiendan la utilidad que podría tener en algún contexto determinado, "select distinct". Por ejemplo, para no ver los valores duplicados que sí se ven aquí, podríamos utilizar "select distinct", ¿cuál es el contexto de esto? Supongan que nos piden, okey, "queremos que revise la base de datos de compradores de "customers" y nos digas en qué países nos están comprando", bueno, si yo muestro esto, puede decir, bien, nos están comprando en Alemania, en México, México, en el Reino Unido, en Suiza, en Alemania, otra vez, no tiene sentido, ¿okey? Ese sería un buen contexto, creo que así lo puedo explicar bien, entonces aquí si nos pidieran eso haríamos "select distinct country", el nombre del campo, clave, sería "país", o sea, "seleccionar, distinct, país, de la tabla customers", "from customers" minúscula, bueno es igual. y ahora podríamos obtener la misma lista que antes. pero más corta, porque ya no hay valores repetidos como pueden ver aquí, okey. Bien, esto nos deja, tienen que recorrer uno por uno para ver si es verdad que no hay valores repetidos, ¿no?, podríamos hacer un hack aquí, podríamos indicar "order by" y bueno, también "customers" por el mismo, por el mismo campo, no, perdón, aquí estoy cometiendo, estoy cometiendo un error, aquí estoy cometiendo un error. Vamos a dejar el tema de ordenar los campos para dentro de nada, de un segundo nada más, okey, para no confundirlos. Bien, bien, ya que estamos aquí y que cometí este spoiler, no lo dejemos para después, porque sería, sería una picardía, ¿verdad?, entonces vamos a ver si podemos concatenar esto, la palabra clave "order by" se usa para ordenar el conjunto de resultados en orden ascendente o descendente, ¿sí? la sintaxis es de poner, por ejemplo, aquí "select distinct country from customers", podríamos ordenar los campos de esta manera, podríamos indicar "order by" y aquí tendríamos que ordenar el nombre de la columna por el que queremos ordenar, por ejemplo, podríamos ordenar, bueno, otra vez por el campo "country", vamos a ponerlo en minúsculas, así, y ahora si tenemos el mismo resultado que hoy, okey, ¿sí? Fíjense nosotros habíamos escrito esto y nos mostraba los campos con la clave "país", es decir, nos mostraba todos los registros, pero sin repetir el país, sin repetir los países, claro, no podíamos ver bien el orden, tendríamos que andar mirando uno por uno. De esta manera, al agregar "order by" y también pedimos que nos ordene por el mismo campo, los ordena de forma ascendente y así, es más fácil chequear que, efectivamente, no tenemos campos repetidos.

Y ya que me adelanté un poco en el tiempo, perdón, y me vine a a la posibilidad de ordenar un campo, también podemos indicar que ese orden sea ascendente o descendente, en realidad, por defecto es ascendente. Si queremos que sea descendente, debemos indicar lo aquí explícitamente y ahora comenzaremos por, bueno, en este caso Venezuela y terminaríamos en Argentina, creo, efectivamente. Bueno, muy bien, vamos a seguir entonces, perdón, que me fui un poco hacia adelante en el futuro, pero me parece que

estaba bien, porque en este caso, en este ejemplo, ¿no?, nos venía bien no poder ordenar los campos y bueno ya han aprendido cómo ordenarlos por otra parte, okey. Vamos a ver qué hace la cláusula “where”, vamos a visitar la cláusula “where”. La cláusula “where”, o “dónde”, se usa para filtrar registros y extraer solamente aquellos que cumplen una condición o cláusula. A ver, podríamos por ejemplo, ¡ah!, esto es importante que lo sepan, la cláusula la condición “where”, no solamente se usa en la, en el listado cuando yo quiero “select”, hacer un “select”, también puedo, como vimos hoy, por ejemplo, borrar o actualizar registros basados en esta cláusula, ¿sí?, por ejemplo, podríamos borrar todos aquellos customers cuyo país sea, bueno, algunos de los países de la tabla, ¿okey?, eso también podríamos hacerlo, no lo vamos a hacer ahora, vamos a seguir con otras consultas, pero sepan que eso se puede hacer, es más, les voy a hacer un ejemplo acá, sin ejecutarlo, por ejemplo “select”, todo, todos los campos, con “customers”, “where”, es decir, ¿dónde?, allí donde el país, el campo país, sea igual a “Germany” y bueno, esto va entre comillas, porque es un texto, ¿no?, y nos va a mostrar todos los customers donde el campo “country” sea igual a “Germany”, ¿okey? Yo podría haber hecho esto también si podría ser un “delete”, “delete from customers where country equals Germany” y esto borraría todos los registros, los mismos que vimos recién, bien, no lo vamos a hacer ahora, porque quiero seguir utilizando la información que está en esta tabla, aunque, bueno, luego, podría volver a importarla, pero no es el caso, creo que con esto les queda claro. Que también podríamos hacer un “update”, okey, con este, con esta misma cláusula.

Y ahora vamos a ver operadores de comparación. Podríamos ver, por ejemplo, los operadores, que les voy a mostrar aquí brevemente, “Y”, “O” o “NOT”, tal como veíamos, por ejemplo, en Javascript, es muy parecido, esto es muy parecido. Por ejemplo, miren, podríamos seleccionar algún ejemplo, a ver aquí, que se nos ocurra sobre la información que tenemos, bueno, podríamos pedir aquí “select”, todo de la tabla “customers”, “where country equals Germany” y vamos a ver alguna otra, a un “OR”, ¿okey?, pongamos un “OR”, ahora “OR country, equals, Argentina” y esto debería darnos la lista de todos los customers, todos los clientes o compradores que residan en Alemania o en Argentina, y aquí los tenemos. Bien. Tenemos los que residen en Alemania o en Argentina, en este caso puntual, si nosotros en vez de “OR” pusieramos “AND” no habría un cambio, perdón, sí habría un cambio, ¿por qué? Bueno, porque aquí le estoy diciendo que tienen que recibir en Alemania y en Argentina al mismo tiempo, cosa que no es posible, o por lo menos no lo es en el contexto de nuestra tabla. Nuestra tabla tiene un solo país en el campo “country”, ¿okey?, o sea, que si ponemos esta condición, no nos va a devolver ningún registro, entonces, si escribimos “OR”, vamos a buscar aquellos que el país sea igual a lo que nosotros escribamos aquí, o aquí, ¿okey? Si ponemos y se tienen que cumplir ambas condiciones, por ejemplo, podríamos decir que vivan en Alemania y que se cumpla alguna otra condición, tendríamos, miren vamos a dar, vamos a verlo ya que estamos aquí, hagamos esto aquí, nos está mostrando todos los que el país es Alemania, ya buscamos que se cumpla también otra condición, bueno, a ver, la ciudad, Stuttgart, Münster, ... , Berlín. Bueno, podríamos indicar a aquellos que vivan en Alemania y que la ciudad además sea Berlín, por ejemplo, o que en el título de contacto sean sales representative. Aquí me lo voy a copiar, por si después no lo digo tal cual, así que aquí podríamos, poco jugar con esto, ¿cierto?, podríamos poner que seleccione todos los clientes cuyo país sea Alemania y “city” equal “Berlín” y esto lo mostraría uno solo cierto, porque había uno solo que cumplía esa condición aquí en la compañía de Alfred Fütterkiste, no bueno, no sé pronunciar el alemán muy bien, perdón, y ¿qué pasa si preguntamos por aquellos que viven en Alemania y cuyo contact title sea “sales representative” o sea

representante de ventas?, creo que ahí encontraríamos más registros que sólo uno, entonces podríamos poner aquí, "contact title", equals, "sales representative", bueno aquí obtenemos unos cuantos más, noten que se cumple esta condición, la del país, pero también esta. Las dos. Bueno, muy bien, y podemos ir concatenando tal cual hacíamos en Javascript o en cualquier lenguaje, estos conectores, ¿okey?, estas condiciones, estos operadores lógicos, el "Y", el "O" o el "NOT", también para indicar aquellos valores que no cumplan una condición puntual. Eso también podemos hacerlo perfectamente muy bien, podríamos probarlo con la sintaxis "NOT", por ejemplo, sigamos con estos y podríamos mostrar que cumplan estas dos condiciones, pero que además, no vivan en Berlín, o que no estén en la ciudad de Berlín, entonces tendría que mostrarnos estos tres campos pero no éste. A ver, vamos a actualizar, vamos a mostrar todos los "customers" donde el país sea igual a Alemania, "contact title" sea igual a "sales representative", pero que además, "AND", vamos a poner aquí, a ver, vamos a poner aquí, cómo podemos escribirlo para que sea prolijo, "NOT, city, equals Berlín", bueno, así es y esto nos muestra tal como preveíamos los tres registros que tienen como título de contacto "representante de ventas", como país de origen Alemania, pero ha quedado afuera aquel cuya ciudad es Berlín y esto puede ser tan complejo como ustedes necesiten, ¿okey?

Aquí tenemos entonces alguna manera de preguntar un poco más en profundidad de hacer nuestras consultas SQL con el comando "select". Recuerden comenzamos con "select", asterisco, "from" la tabla, y llegamos a algo como esto, y podríamos hacerlo más o menos complejo, de acuerdo a nuestras necesidades.