



CURSO SQL



MySQL®



Para acceder al curso en YouTube llamado pildorasinformaticas, done accederás a los videos y los archivos para el curso.

PERE MANEL VERDUGO ZAMORA

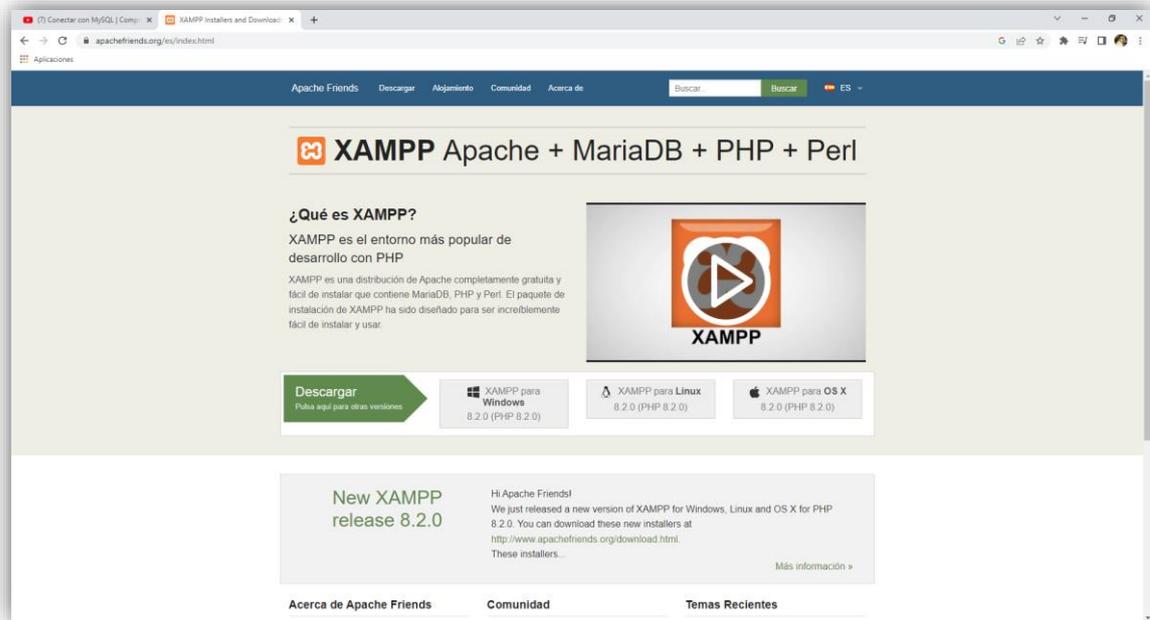
www.peremanelv.com
pereverdugo@gmail.com

Contenido

Para este curso que necesitamos	2
Capítulo 1.- Presentación	4
Capítulo 2.- Introducción a SQL.	5
¿Qué es S.Q.L?	5
Capítulo 3.- Cláusulas y operadores.....	12
Capítulo 4.- Cláusula Order By. Ordenador registros.....	25
Capítulo 5.- Consulta de agrupación o totales	34
Capítulo 6.- Consultas de cálculo.....	43
Capítulo 7.- Consultas Multitabla I	51
Capítulo 8.- Consultas Multitabla II Inner Join	57
Capítulo 9.- Consultas Multitabla III. Left Join y Right Join.....	61
Capítulo 10.- Subconsultas I	66
Capítulo 11.- Subconsultas II	71
Capítulo 12.- Consultas de acción I	75
Capítulo 13.- Consultas de acción II Eliminación	80
Capítulo 14.- Consultas de acción III. Consulta de datos anexados.....	87
Capítulo 15.- Referencias cruzadas (Access)	93
Capítulo 16.- Creación de tablas.....	106
Capítulo 17.- Agregar, eliminar y modificar campos.....	113
Capítulo 18.- Índices.....	121
Capítulo 19.- Eliminación de Índices	131
Capítulo 20.- Triggers I (Disparadores).....	134
Capítulo 21.- Triggers II (Disparadores).....	140
Capítulo 22.- Triggers III (Disparadores).....	146
Capítulo 23.- Procedimientos almacenados I	153
Capítulo 24.- Procedimientos y triggers.....	159
Capítulo 25.- Vistas (Final curso)	165
Capítulo 26.- Ejercicio final I (Modelos)	170
Capítulo 27.- Ejercicio Final II (Creando la BD)	171
Capítulo 28.- Ejercicio Final III (Creando la tabla CALIFICACIONES).....	172
Capítulo 29.- Ejercicio Final IV (Insertando Datos)	173
Capítulo 30.- Ejercicio Final V (Insertando datos tabla CALIFICACIONES	175
Capítulo 31.- Ejercicio Final VI (Alias o sobrenombres)	178
Capítulo 32.- Ejercicio Final VII (UNIONES ENTRE TABLAS)	179

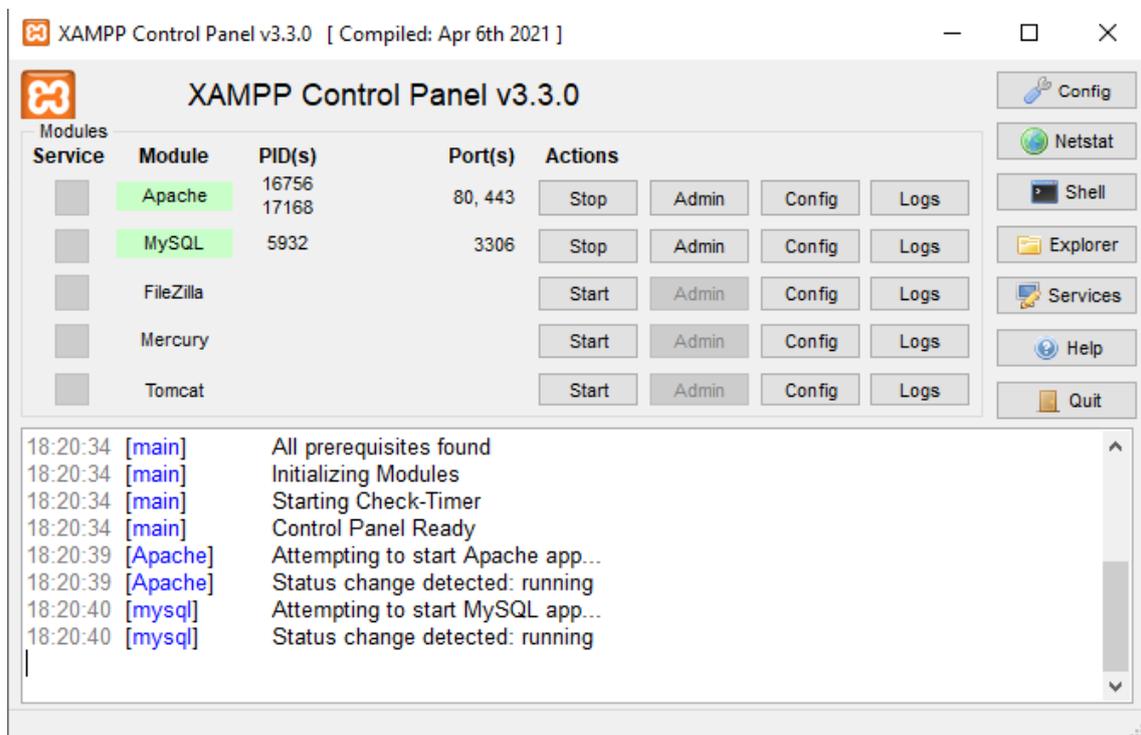
Para este curso que necesitamos
Para poder realizar esta práctica tendrás que instalar XAMPP.

<https://www.apachefriends.org/es/index.html>



Después de descargarlo procederemos a su instalación.

Cuando lo ejecutemos:

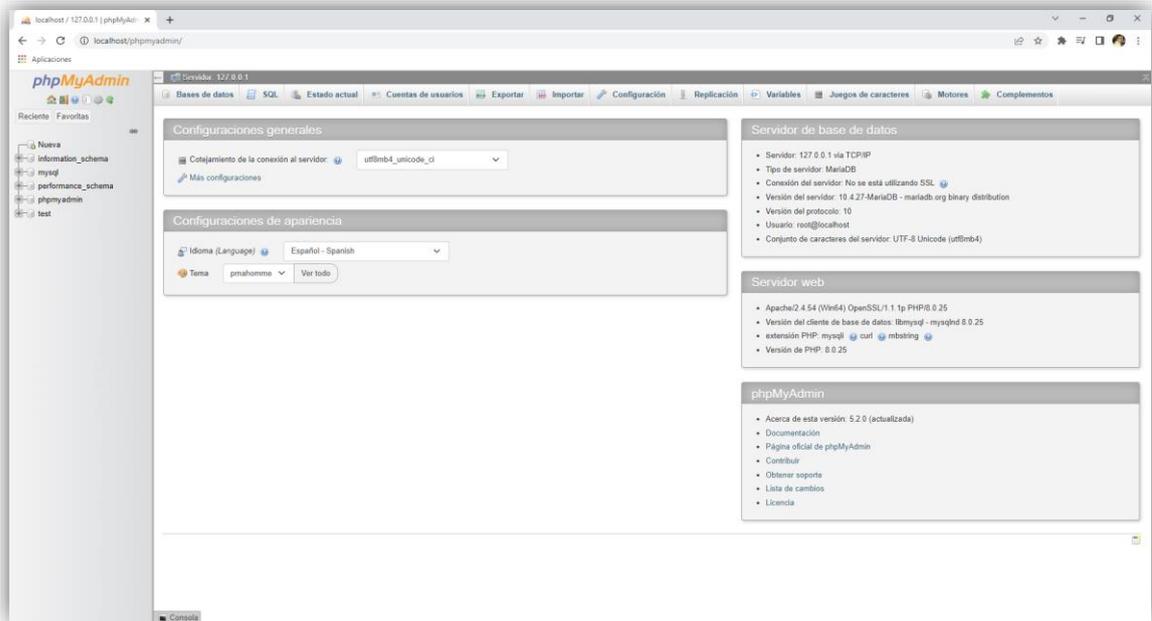


Activaremos Apache y MySql.

Ya podemos minimizar esta ventana pero no cerrarla.

Ejecutaremos el navegador y accederemos a la siguiente dirección.

<http://localhost/phpmyadmin/>



Capítulo 1.- Presentación

¿A quién va dirigido este curso?

A personas con:

- Conocimiento acerca de que es una BBDD.
- Hayan manejado algún gestor de BBDD con anterioridad.
- Utilice alguna tecnología que requiera conocer el lenguaje SQL.

Temario:

- Características del lenguaje SQL. Utilidad del lenguaje.
- Comandos SQL. Grupos de comandos.
- Consultas de selección.
 - Consultas multi-Tabla.
 - Consultas de agrupación o resumen.
 - Consultas de cálculo.
 - Subconsultas.
- Consultas de acción
 - Creación de tablas.
 - Actualización.
 - Eliminación.
 - Datos anexados.
- Consultas de referencias cruzadas.
- Consulta de definición de datos.
 - Tipo de datos.
 - Índices.
 - Integridad referencial.

Capítulo 2.- Introducción a SQL.

En este capítulo veremos...

- Qué es SQL.
- Estándar SQL.
- Grupos de comandos SQL.
- Primera sentencia SQL.

¿Qué es S.Q.L?

- Structured Query Language: Lenguaje estructurado de consultas. Inventado por IBM.
- Es un lenguaje para interactuar con BBDD relacionales.

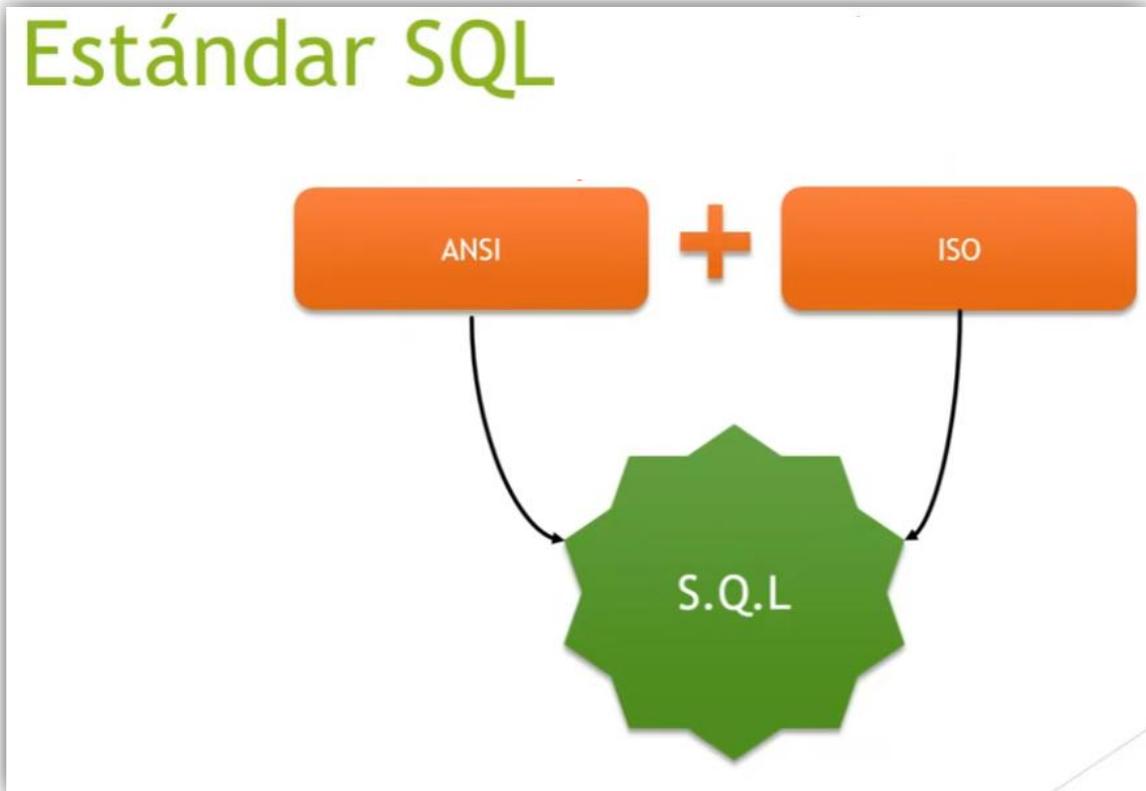


Necesitamos un gestor de BBDD encargado de interactuar con la base de datos.



Que tienen en común estos gestores de BBDD, pues que adoptan el estándar SQL como sistema para interactuar con la Base de Datos.

Sql no es considerado un lenguaje de programación propiamente dicho, porque carece de las estructuras que tienen los lenguajes de programación, es un lenguaje de acceso de datos.



Comandos DDL: Sirve para crear y modificar la estructura de una base de datos.

Comandos DML: Se utiliza para seleccionar registros de una tabla, consultas, además de insertar registros, eliminación de registros, etc. (Consultas de selección y de acción).

Comandos DCL: Se utiliza para proporcionar seguridad en la base de datos.

Comandos TCL: Se utiliza para la gestión de los cambios en los datos.

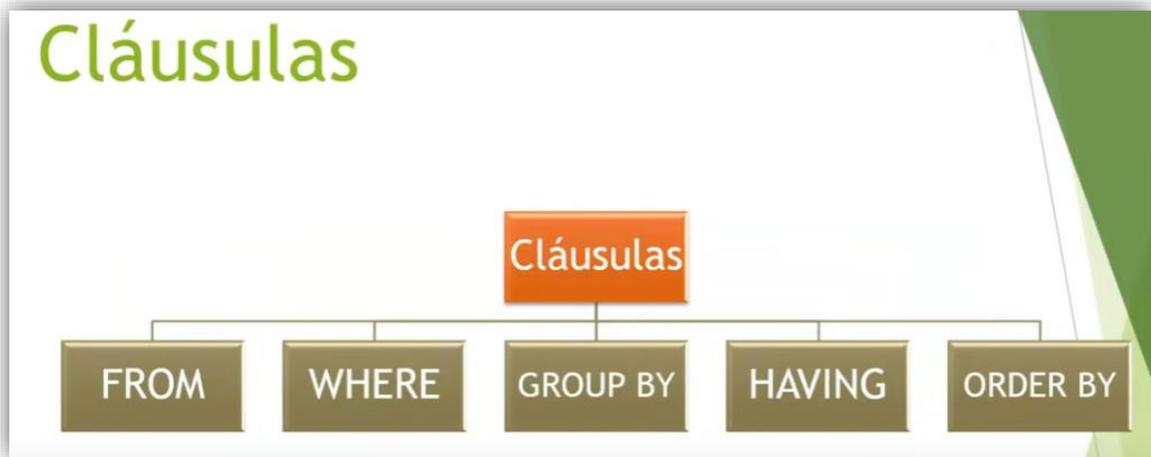
Los comandos de DDL son: CREATE, ALTER, DROP y TRUNCATE.

Los comandos de DML son: SELECT, INSERT, UPDATE y DELETE.

Los comandos de DCL son: GRANT y REVOKE.

Los comandos de TCL son: COMMIT, ROLLBACK y SAVEPOINT.

También tiene cláusulas:



¿Qué son las instrucciones?



Uniendo todo esto como si fuese una frase en inglés construimos una frase SQL.

No es necesario que una instrucción SQL lleve los 4 componentes.

Vamos con la práctica:

En el siguiente enlace podrás descargar una base de datos para las primeras prácticas.

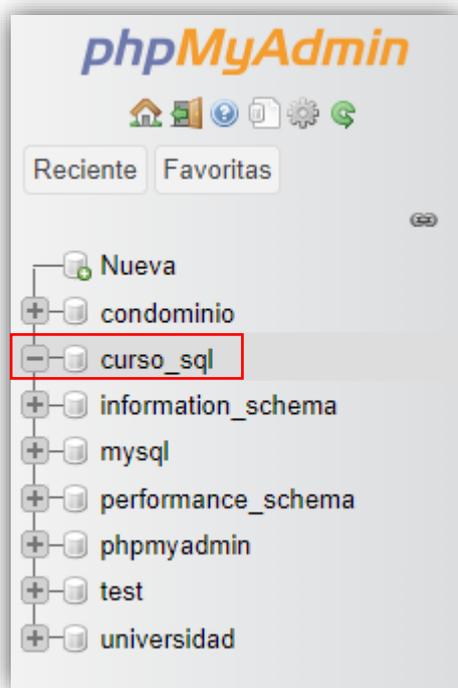
Para descargar la base de datos selección [aquí](#):

Crear una carpeta para el curso y en ella lo descomprimes.





Para ello vamos a crear una base de datos, la vamos a llamar Curso_SQL, seguido del botón crear.



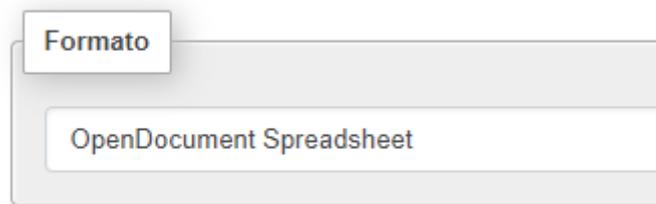
Ya le hemos creado.

La seleccionamos desde al izquierda y a continuación seleccionamos la pestaña importar.

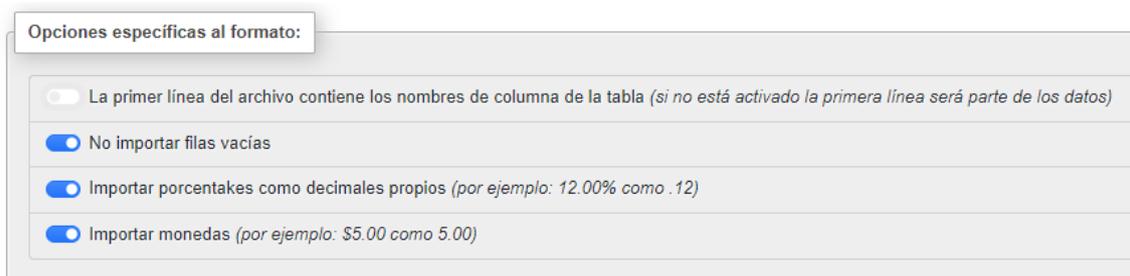


Seleccionamos el archivo CLIENTES.ods.

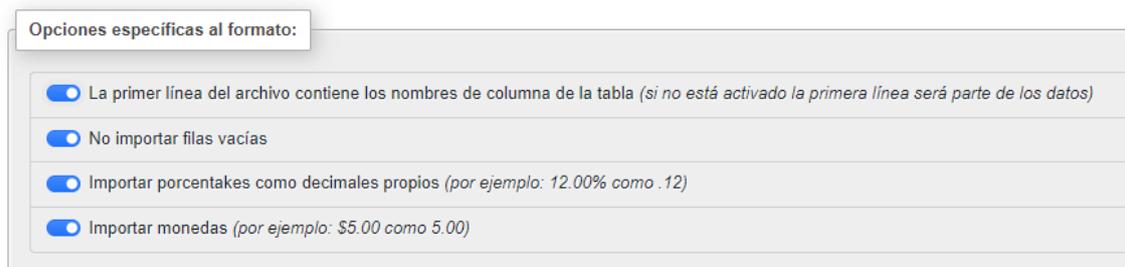
En formato tiene que aparecer



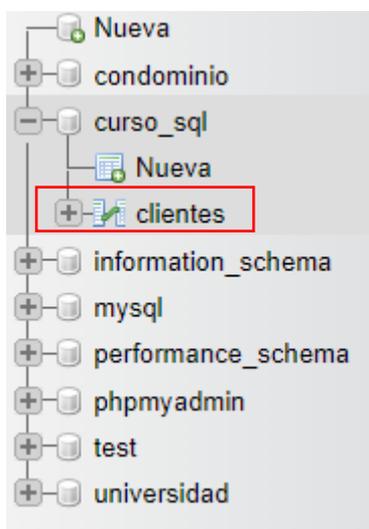
Si no elegimos el menú desplegable para seleccionar esta opción.



Habilitar la primera opción que dice 'Contiene los nombres de columna de la tabla'.



Le damos a importar.



Si todo ha ido bien tendremos la tabla clientes en la base de datos curso_sql.

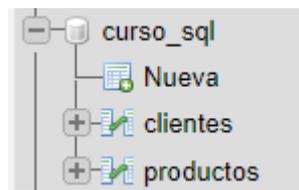
Si la seleccionamos nos mostrará todos los registros.

1 > >> Mostrar todo | Número de filas: 25 | Filtrar filas:

Opciones extra

CÓDIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACIÓN	TELÉFONO	RESPONSABLE	HISTORIAL
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTÍNEZ	NULL
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA	NULL
CT03	EL ESPAÑOLITO	MOTORES 34	BARCELONA	934565343	ANA FERNÁNDEZ	NULL
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GÓMEZ	NULL
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ	NULL
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	912649987	ELÍAS PÉREZ	NULL
CT08	JUGUETERÍA SUÁREZ	PARIS 123	BARCELONA	933457866	JUAN GARCÍA	NULL
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ	NULL
CT10	FERETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	914354866	MANUEL MENÉNDEZ	NULL
CT11	JUGUETES MARTÍNEZ	VIA LAYETANA 245	BARCELONA	936628554	FRANCISCO CUEVAS	NULL
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO	NULL
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL	NULL
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	913299475	ANA JIMÉNEZ	NULL
CT15	EXCLUSIVAS FERNÁNDEZ	LLOBREGAT 250	BARCELONA	939558365	LUISA FERNÁNDEZ	NULL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO	NULL
CT17	BAZAR FRANCISCO	CARMEN 45	ZAMORA	980495288	CARLOS BELTRÁN	NULL

Lo vamos a repetir con productos.



Teniendo seleccionada la base de datos curso_sql nos vamos a la pestaña SQL.

phpMyAdmin - Servidor: 127.0.0.1 - Base de datos: curso_sql

Reciente Favoritas

- Nueva
- condominio
- curso_sql
 - Nueva
 - clientes
 - productos
- information_schema
- mysql
- performance_schema
- phpmyadmin
- test
- universidad

Filtros

Que contengan la palabra:

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> clientes	Examinar Estructura Buscar Insertar Vaciar Eliminar	38	InnoDB	utf8_general_ci	16.0 KB	-
<input type="checkbox"/> productos	Examinar Estructura Buscar Insertar Vaciar Eliminar	40	InnoDB	utf8_general_ci	16.0 KB	-
2 tablas	Número de filas	78	InnoDB	utf8mb4_general_ci	32.0 KB	0 B

Seleccionar todo Para los elementos que están marcados:

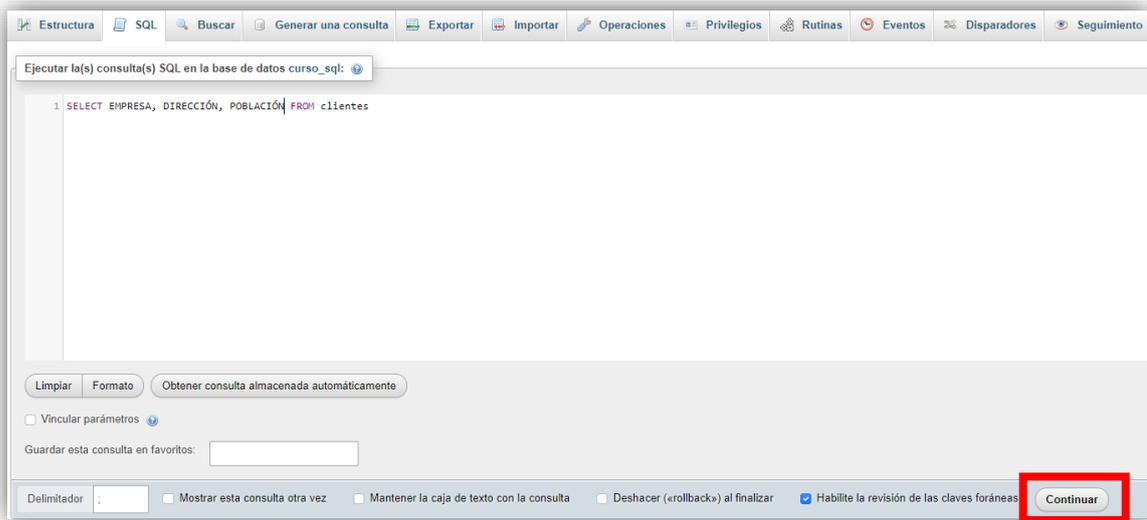
Ejecutar la(s) consulta(s) SQL en la base de datos curso_sql:

1 |

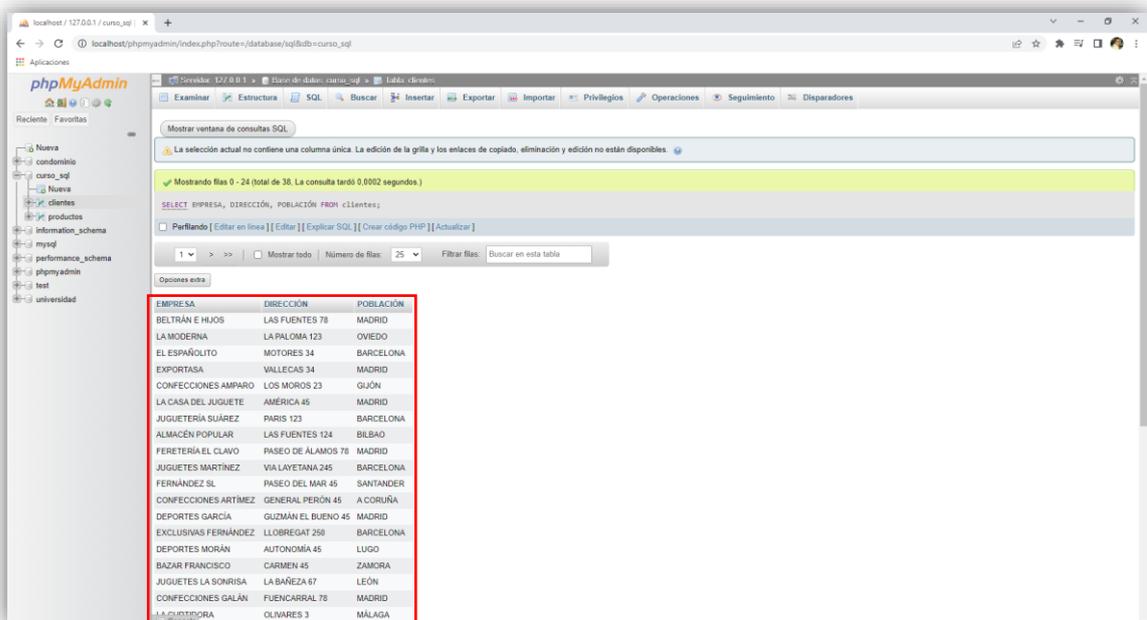
Limpiar Formato Obtener consulta almacenada automáticamente

Ya podemos empezar a escribir las instrucciones SQL.

```
1 SELECT EMPRESA, DIRECCIÓN, POBLACIÓN FROM clientes
```



Seleccionaremos el botón 'Continuar'.



La consulta se ejecuta mostrando el siguiente resultado.

Capítulo 3.- Cláusulas y operadores

En este capítulo veremos ...

- Cláusulas SQL
- Operadores
 - Comparación
 - Lógicos

Cláusulas

Cláusula	Descripción
FROM	Especifica la tabla de la que se quieren obtener los registros
WHERE	Especifica las condiciones o criterios de los registros seleccionados
GROUP BY	Para agrupar los registros seleccionados en función de un campo
HAVING	Especifica las condiciones o criterios que deben cumplir los grupos
ORDER BY	Ordena los registros seleccionados en función de un campo

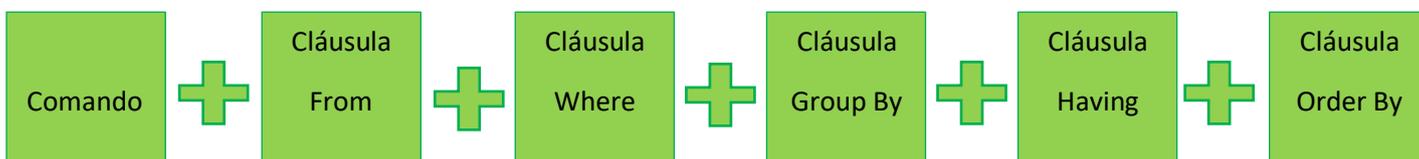
Operadores de comparación

Operador	Significado
<	Menor que
>	Mayor que
=	Igual que
>=	Mayor o igual que
<=	Menor o igual que
<>	Distinto que
BETWEEN	Entre. Utilizado para especificar rangos de valores
LIKE	Cómo. Utilizado con caracteres comodín (? *)
In	En. Para especificar registros en un campo en concreto

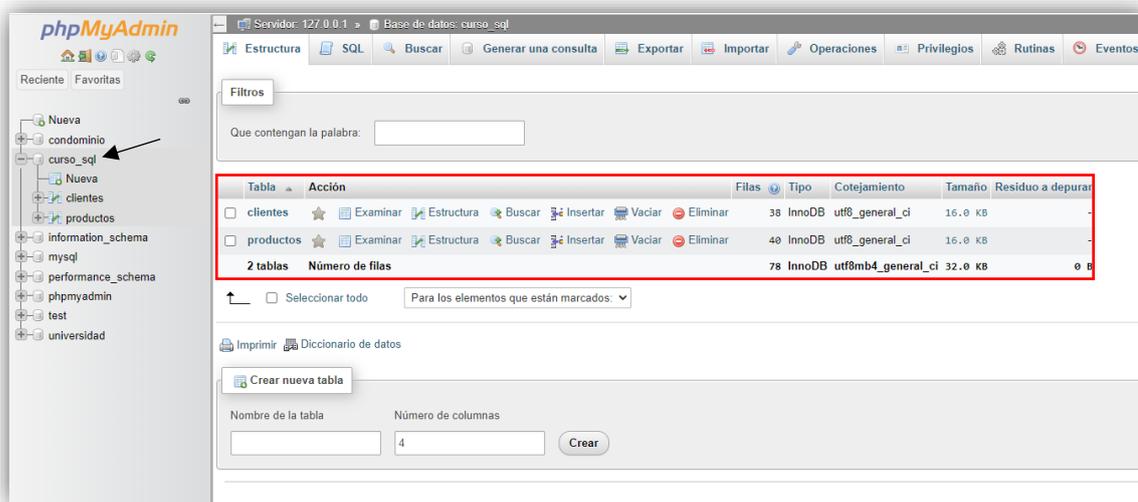
Operadores lógicos

Operador	Significado
AND	Y lógico
OR	O lógico
NOT	Negación lógica

Orden de escritura:

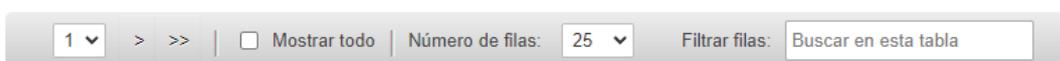


Nos vamos a Sql.



Si seleccionamos la base de datos nos muestra las dos tablas que importamos en el capítulo anterior.

Vamos a seleccionar la tabla producto.



Opciones extra

CÓDIGOARTÍCULO SECCIÓN NOMBREARTÍCULO PRECIO FECHA IMPORTADO PAÍSDEORIGEN FOTO

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR01	FERRETERÍA	DESTORNILLADOR	6,63	22/10/2000	FALSO	ESPAÑA	NULL
AR02	CONFECCIÓN	TRAJE CABALLERO	284,58	11/03/2002	VERDADERO	ITALIA	NULL
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159,45	26/05/2002	VERDADERO	MARRUECOS	NULL
AR04	DEPORTES	RAQUETA TENIS	93,47	20/03/2000	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60,00	13/09/2000	VERDADERO	USA	NULL
AR07	CONFECCIÓN	SERRUCHO	30,20	23/03/2001	VERDADERO	FRANCIA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103,34	11/04/2000	VERDADERO	JAPÓN	NULL
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174,23	10/01/2000	VERDADERO	MARRUECOS	NULL
AR10	JUGUETERÍA	CONSOLA VIDEO	442,54	24/09/2002	VERDADERO	USA	NULL
AR11	CERÁMICA	TUBOS	168,43	04/02/2000	VERDADERO	CHINA	NULL
AR12	FERRETERÍA	LLAVE INGLESA	24,40	23/05/2001	VERDADERO	USA	NULL
AR13	CONFECCIÓN	CAMISA CABALLERO	67,13	11/08/2002	FALSO	ESPAÑA	NULL
AR14	JUGUETERÍA	TREN ELÉCTRICO	1.505,38	03/07/2001	VERDADERO	JAPÓN	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54,09	07/06/2000	VERDADERO	CHINA	NULL
AR16	FERRETERÍA	ALICATES	6,74	17/04/2000	VERDADERO	ITALIA	NULL
AR17	JUGUETERÍA	MUÑECA ANDADORA	105,06	04/01/2001	FALSO	ESPAÑA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46,73	02/02/2001	VERDADERO	SUECIA	NULL

Podemos observar los nombres de los campos.

Quiero realizar una consulta de aquellos artículos que pertenecen a la sección de cerámica.

Nos vamos a la pestaña SQL.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO FROM productos WHERE SECCIÓN = "CERÁMICA"
```

Le damos al botón continuar.

NOMBREARTÍCULO	SECCIÓN	PRECIO
TUBOS	CERÁMICA	168,43
PLATO DECORATIVO	CERÁMICA	54,09
JUEGO DE TE	CERÁMICA	43,27
CENICERO	CERÁMICA	19,75
MACETA	CERÁMICA	29,04
JARRA CHINA	CERÁMICA	127,77

Estos son los registros que cumplen con la condición.

Ahora queremos conseguir los artículos de la sección de "CERÁMICA" y "DEPORTES".

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO FROM productos WHERE SECCIÓN = "CERÁMICA" OR SECCIÓN = "DEPORTES"
```

Le damos al botón continuar.

NOMBREARTÍCULO	SECCIÓN	PRECIO
RAQUETA TENIS	DEPORTES	93,47
MANCUERNAS	DEPORTES	60,00
TUBOS	CERÁMICA	168,43
PLATO DECORATIVO	CERÁMICA	54,09
PISTOLA OLÍMPICA	DEPORTES	46,73
JUEGO DE TE	CERÁMICA	43,27
CENICERO	CERÁMICA	19,75
BALÓN RUGBY	DEPORTES	111,64
BALÓN BALONCESTO	DEPORTES	75,27
BALÓN FÚTBOL	DEPORTES	43,91
CRONÓMETRO	DEPORTES	439,18
MACETA	CERÁMICA	29,04
CAÑA DE PESCA	DEPORTES	270,00
JARRA CHINA	CERÁMICA	127,77
BOTA ALPINISMO	DEPORTES	144,00
PALAS DE PING PONG	DEPORTES	21,60

Queremos ver todos los campos de la tabla artículos de deportes y además de la sección americana.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = "DEPORTES" AND PAÍSDEORIGEN = "USA"
```

Le damos al botón continuar.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93,47	20/03/2000	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60,00	13/09/2000	VERDADERO	USA	NULL
AR24	DEPORTES	BALÓN RUGBY	111,64	11/11/2000	VERDADERO	USA	NULL
AR32	DEPORTES	CRONÓMETRO	439,18	03/01/2002	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270,00	14/02/2000	VERDADERO	USA	NULL

Quiero hacer una consulta que muestre los artículos cuyo precio sea superior a 300 €.

```
1 SELECT * FROM PRODUCTOS WHERE PRECIO > 300
```

En definitiva esta consulta parece que está bien pero si la consultamos.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR10	JUGUETERÍA	CONSOLA VIDEO	442,54	24/09/2002	VERDADERO	USA	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522,69	10/07/2001	VERDADERO	ITALIA	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500.000,00	05/04/2002	VERDADERO	ITALIA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360,07	03/05/2001	VERDADERO	MARRUECOS	NULL
AR32	DEPORTES	CRONÓMETRO	439,18	03/01/2002	VERDADERO	USA	NULL

Algunos artículos no salen.

Si nos vamos a la pestaña estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 CÓDIGOARTÍCULO	varchar(4)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	2 SECCIÓN	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	3 NOMBREARTÍCULO	varchar(19)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4 PRECIO	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	5 FECHA	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	6 IMPORTADO	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7 PAÍSDEORIGEN	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8 FOTO	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más

Tanto el campo PRECIO como FECHA son de tipo varchar de tipo texto.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 CÓDIGOARTÍCULO	varchar(4)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	2 SECCIÓN	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	3 NOMBREARTÍCULO	varchar(19)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4 PRECIO	decimal(10,0)			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	5 FECHA	date			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	6 IMPORTADO	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7 PAÍSDEORIGEN	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8 FOTO	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más

Al PRECIO lo cambiamos a decimal y FECHA de tipo Date.

Vamos a realizar la consulta de nuevo.

```
1 SELECT * FROM PRODUCTOS WHERE PRECIO > 300
```

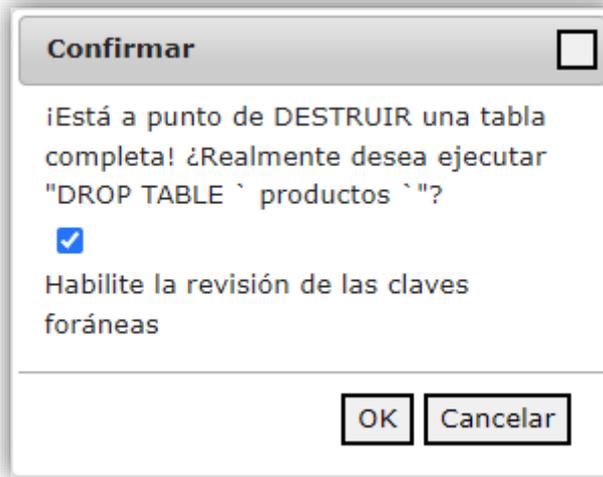
Ahora este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR10	JUGUETERÍA	CONSOLA VIDEO	442	0000-00-00	VERDADERO	USA	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522	0000-00-00	VERDADERO	ITALIA	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500	0000-00-00	VERDADERO	ITALIA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360	0000-00-00	VERDADERO	MARRUECOS	NULL
AR32	DEPORTES	CRONÓMETRO	439	0000-00-00	VERDADERO	USA	NULL

Otra solución sería haberse preocupado por los datos en origen.
 Nos vamos a la base de datos y eliminamos la tabla de PRODUCTOS.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> clientes	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	38	InnoDB	utf8_general_ci	16.0 KB	-
<input checked="" type="checkbox"/> productos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	40	InnoDB	utf8_general_ci	16.0 KB	-
2 tablas	Número de filas	78	InnoDB	utf8mb4_general_ci	32.0 KB	0 B

La seleccionamos y la eliminamos.



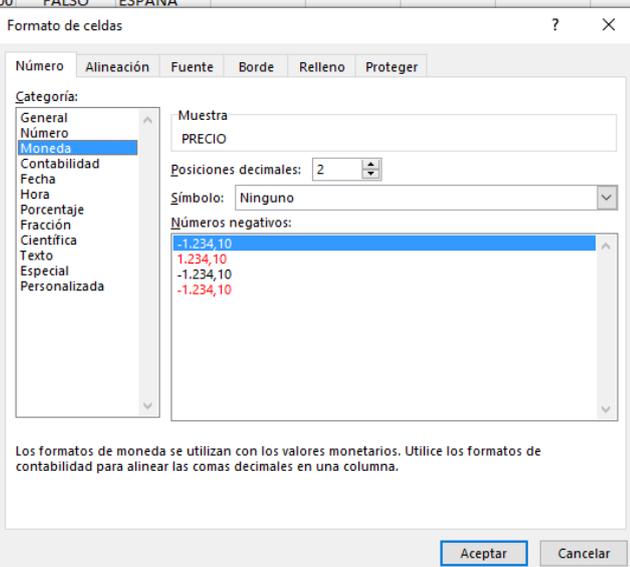
Nos pedirá confirmación a lo que contestaremos OK.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> clientes	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	38	InnoDB	utf8_general_ci	16.0 KB	-
1 tabla(s)	Número de filas	38	InnoDB	utf8mb4_general_ci	16 KB	0 B

La tabla ya no está.

Antes de volverlo a importar tenemos que abrirlo con la hoja de calculo y darle su formato.

	A	B	C	D	E	F	G	H	I	J	K	L
1	CÓDIGOART	SECCIÓN	NOMBREART	PRECIO	FECHA	IMPORTADO	PAÍS	DEORIGI	FOTO			
2	AR01	FERRETERÍA	DESTORNILL	6,63	22/10/2000	FALSO	ESPAÑA					
3	AR02	CONFECCIÓN	TRAJE CABAL	284,58	11/03/20							
4	AR03	JUGUETERÍA	COCHE TELE	159,45	26/05/20							
5	AR04	DEPORTES	RAQUETA TE	93,47	20/03/20							
6	AR06	DEPORTES	MANCUERNA	60,00	13/09/20							
7	AR07	CONFECCIÓN	SERRUCHO	30,20	23/03/20							
8	AR08	JUGUETERÍA	CORREPASIL	103,34	11/04/20							
9	AR09	CONFECCIÓN	PANTALÓN S	174,23	10/01/20							
10	AR10	JUGUETERÍA	CONSOLA VI	442,54	24/09/20							
11	AR11	CERÁMICA	TUBOS	168,43	04/02/20							
12	AR12	FERRETERÍA	LLAVE INGLE	24,40	23/05/20							
13	AR13	CONFECCIÓN	CAMISA CAB	67,13	11/08/20							
14	AR14	JUGUETERÍA	TREN ELÉCTR	1.505,38	03/07/20							
15	AR15	CERÁMICA	PLATO DECO	54,09	07/06/20							
16	AR16	FERRETERÍA	ALICATES	6,74	17/04/20							
17	AR17	JUGUETERÍA	MUÑECA AN	105,06	04/01/20							
18	AR18	DEPORTES	PISTOLA OLÍ	46,73	02/02/20							
19	AR19	CONFECCIÓN	BLUSA SRA.	101,06	18/03/20							
20	AR20	CERÁMICA	JUEGO DE TE	43,27	15/01/20							
21	AR21	CERÁMICA	CENICERO	19,75	02/07/20							
22	AR22	FERRETERÍA	MARTILLO	11,40	04/09/20							
23	AR23	CONFECCIÓN	CAZADORA F	522,69	10/07/20							
24	AR24	DEPORTES	BALÓN RUGE	111,64	11/11/20							
25	AR25	DEPORTES	BALÓN BALC	75,27	25/06/20							
26	AR26	JUGUETERÍA	FUERTE DE S	143,70	25/11/20							
27	AR27	CONFECCIÓN	ABRIGO CAB	500.000,00	05/04/2002	VERDADERO	ITALIA					



A PRECIO le damos el formato moneda.

	A	B	C	D	E	F	G	H	I	J	K	L
1	CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO				
2	AR01	FERRETERÍA	DESTORNILLO	6,63	22/10/2000	FALSO	ESPAÑA					
3	AR02	CONFECCIÓN	TRAJE CABALLERO	284,58	11/03/2002	VERDADERO	ITALIA					
4	AR03	JUGUETERÍA	COCHE TELECOMANDO	159,45	26/05/2002	VERDADERO	MARRUECOS					
5	AR04	DEPORTES	RAQUETA TENIS	93,47	20/03/2000	VERDADERO	MARRUECOS					
6	AR06	DEPORTES	MANCUERNAS	60,00	13/09/2000	VERDADERO	MARRUECOS					
7	AR07	CONFECCIÓN	SERRUCHO	30,20	23/03/2001	VERDADERO	MARRUECOS					
8	AR08	JUGUETERÍA	CORREPASILLO	103,34	11/04/2000	VERDADERO	MARRUECOS					
9	AR09	CONFECCIÓN	PANTALÓN SUEÑO	174,23	10/01/2000	VERDADERO	MARRUECOS					
10	AR10	JUGUETERÍA	CONSOLA VIDEO	442,54	24/09/2002	VERDADERO	MARRUECOS					
11	AR11	CERÁMICA	TUBOS	168,43	04/02/2000	VERDADERO	MARRUECOS					
12	AR12	FERRETERÍA	LLAVE INGLESA	24,40	23/05/2001	VERDADERO	MARRUECOS					
13	AR13	CONFECCIÓN	CAMISA CAMPER	67,13	11/08/2002	VERDADERO	MARRUECOS					
14	AR14	JUGUETERÍA	TREN ELÉCTRICO	1.505,38	03/07/2001	VERDADERO	MARRUECOS					
15	AR15	CERÁMICA	PLATO DECO	54,09	07/06/2000	VERDADERO	MARRUECOS					
16	AR16	FERRETERÍA	ALICATES	6,74	17/04/2000	VERDADERO	MARRUECOS					
17	AR17	JUGUETERÍA	MUÑECA ANIMADA	105,06	04/01/2001	VERDADERO	MARRUECOS					
18	AR18	DEPORTES	PISTOLA OLÍMPICA	46,73	02/02/2001	VERDADERO	MARRUECOS					
19	AR19	CONFECCIÓN	BLUSA SRA.	101,06	18/03/2000	VERDADERO	MARRUECOS					
20	AR20	CERÁMICA	JUEGO DE TETRIS	43,27	15/01/2001	VERDADERO	MARRUECOS					
21	AR21	CERÁMICA	CENICERO	19,75	02/07/2001	VERDADERO	MARRUECOS					
22	AR22	FERRETERÍA	MARTILLO	11,40	04/09/2001	VERDADERO	MARRUECOS					
23	AR23	CONFECCIÓN	CAZADORA FUR	522,69	10/07/2001	VERDADERO	MARRUECOS					
24	AR24	DEPORTES	BALÓN RUGBY	111,64	11/11/2000	VERDADERO	MARRUECOS					
25	AR25	DEPORTES	BALÓN BALONCESTO	75,27	25/06/2001	VERDADERO	MARRUECOS					
26	AR26	JUGUETERÍA	FUERTE DE SUEÑO	143,70	25/11/2000	VERDADERO	MARRUECOS					
27	AR27	CONFECCIÓN	ABRIGO CAMPER	500.000,00	05/04/2002	VERDADERO	MARRUECOS					
28	AR28	DEPORTES	BALÓN FÚTBOL	43,91	04/07/2002	VERDADERO	MARRUECOS					
29	AR29	CONFECCIÓN	ABRIGO SRA.	360,07	03/05/2001	VERDADERO	MARRUECOS					
30	AR30	FERRETERÍA	DESTORNILLO	9,06	20/02/2002	VERDADERO	FRANCIA					
31	AR31	JUGUETERÍA	PISTOLA COM	57,25	15/04/2001	FALSO	ESPAÑA					

Formato de celdas

Categoría: **Personalizada**

Muestra: FECHA

Tipo: aaaa-mm-dd

Eliminar

Escriba el código de formato de número, usando como punto de partida uno de los códigos existentes.

Aceptar Cancelar

A FECHA le damos el formato personalizado aaaa-mm-dd, ya que primero es el año a continuación el mes y por último el día.

Y guardamos los cambios.

Volvemos a importar la tabla en su correspondiente base de datos.

Si seleccionamos la tabla y seleccionamos la pestaña estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	CÓDIGOARTÍCULO	varchar(4)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	2	SECCIÓN	varchar(10)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	3	NOMBREARTÍCULO	varchar(19)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4	PRECIO	decimal(10,4)		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	5	FECHA	varchar(10)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	6	IMPORTADO	varchar(9)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7	PAÍSDEORIGEN	varchar(9)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8	FOTO	varchar(10)	utf8_general_ci	Sí	NULL			Cambiar Eliminar Más

El precio lo ha cambiado pero la fecha, seleccionaremos cambiar.

Nombre	Tipo	Longitud/Valores	Predeterminado
FECHA	DATE	10	NULL

Seleccionar desde las columnas centrales

Seleccionamos el tipo DATE.

Ahora vamos a ver la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 CÓDIGOARTÍCULO	varchar(4)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	2 SECCIÓN	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	3 NOMBREARTÍCULO	varchar(19)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4 PRECIO	decimal(10,4)			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	5 FECHA	date			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	6 IMPORTADO	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7 PAÍSDEORIGEN	varchar(9)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8 FOTO	varchar(10)	utf8_general_ci		Sí	NULL			Cambiar Eliminar Más

Ya hemos cambiado el tipo.

Vamos a realizar una consulta de los artículos desde Marzo y Abril del 2000.

```
1 SELECT * FROM PRODUCTOS WHERE FECHA BETWEEN '2000-03-01' AND '2000-04-30'
```

Le damos a continuar.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPÓN	NULL
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA	NULL
AR19	CONFECCIÓN	BLUSA SRA.	101.0566	2000-03-18	VERDADERO	CHINA	NULL

Hay otra forma de hacerlo.

```
1 SELECT * FROM PRODUCTOS WHERE FECHA >= '2000-03-01' AND FECHA <= '2000-04-30'
```

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPÓN	NULL
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA	NULL
AR19	CONFECCIÓN	BLUSA SRA.	101.0566	2000-03-18	VERDADERO	CHINA	NULL

El resultado es el mismo.

Te adjunto unos ejercicios prácticos para que puedas practicar.

1. Realizar una consulta que muestre los campos “Empresa” y “Población” de la tabla “Clientes”.
2. Realizar una consulta que muestre los artículos d la sección “Cerámica”.
3. Realizar una consulta que muestre los productos de la sección “Deportes” cuyo precio esté entre 100 y 200 €. En la consulta solo se mostrarán los campos “Nombre de artículo” y “Precio”.
4. Realizar una consulta que muestre los productos cuyo país no sea España.
5. Realizar una consulta que muestre los artículos españoles de la sección “Deportes” o aquellos cuyo precio sea superior a 350 € independientemente de cual sea su sección o país de origen.
6. Realizar una consulta que muestre los productos cuya fecha esté entre 1/05/2001 y 15/12/2001. En la consulta solo se visualizarán los campos “Nombre de artículo”, “Sección” y “Fecha”.

SOLUCIÓN

1. Realizar una consulta que muestre los campos “Empresa” y “Población” de tabla “Clientes”.

```
SELECT EMPRESA, POBLACIÓN FROM CLIENTES
```

1 > >> | Número de filas: 25

+ Opciones

EMPRESA	POBLACIÓN
BELTRÁN E HIJOS	MADRID
LA MODERNA	OVIEDO
EL ESPANOLITO	BARCELONA
EXPORTASA	MADRID
CONFECCIONES AMPARO	GIJON
LA CASA DEL JUGUETE	MADRID
JUGUETERÍA SUÁREZ	BARCELONA
ALMACÉN POPULAR	BILBAO
FERRETERÍA EL CLAVO	MADRID
JUGUETES MARTINEZ	BARCELONA
FERNANDEZ SL	SANTANDER
CONFECCIONES ARTIMEZ	A CORUÑA
DEPORTES GARCÍA	MADRID
EXCLUSIVAS FERNÁNDEZ	BARCELONA
DEPORTES MORAN	LUGO
BAZAR FRANCISCO	ZAMORA
JUGUETES LA SONRISA	LEÓN
CONFECCIONES GALAN	MADRID
LA CURTIDORA	MÁLAGA
LÍNEA JOVEN	SEVILLA
BAZAR EL BARAT	BARCELONA
EL PALACIO DE LA MODA	MADRID
SÁEZ Y CÍA	SEVILLA
DEPORTES EL MADRILEÑO	ZARAGOZA
FERRETERÍA LA ESCOBA	MADRID

- Realizar una consulta que muestre los artículos d la sección “Cerámica”.

✓ Mostrando filas 0 - 5 (total de 6, La consulta tardó 0.0010 seg)

```
SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'CERÁMICA'
```

Número de filas: 25

+ Opciones

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAISDEORIGEN	FOTO
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL

- Realizar una consulta que muestre los productos de la sección “Deportes” cuyo precio esté entre 100 y 200 €. En la consulta solo se mostrarán los campos “Nombre de artículo” y “Precio”.

```
SELECT NOMBREARTÍCULO, PRECIO FROM PRODUCTOS WHERE SECCIÓN='DEPORTES' AND PRECIO BETWEEN 100 AND 200
```

Número de filas: 25

+ Opciones

NOMBREARTÍCULO	PRECIO
BALÓN RUGBY	111.6440
BOTA ALPINISMO	144.0000

4. Realizar una consulta que muestre los productos cuyo país no sea España.

```
SELECT * FROM PRODUCTOS WHERE PAISDEORIGEN<>'ESPAÑA'
```

1 > >> | Número de filas: 25

+ Opciones

CÓDIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAISDEORIGEN	FOTO
AR02	CONFECCION	TRAJE CABALLERO	284.5769	2002-03-11	VERDADERO	ITALIA	NULL
AR03	JUGUETERIA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	VERDADERO	MARRUECOS	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR07	CONFECCION	SERRUCHO	30.2045	2001-03-23	VERDADERO	FRANCIA	NULL
AR08	JUGUETERIA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPON	NULL
AR09	CONFECCION	PANTALON SEÑORA	174.2310	2000-01-10	VERDADERO	MARRUECOS	NULL
AR10	JUGUETERIA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA	NULL
AR11	CERAMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR12	FERRETERIA	LLAVE INGLESA	24.3986	2001-05-23	VERDADERO	USA	NULL
AR14	JUGUETERIA	TREN ELECTRICO	1505.3766	2001-07-03	VERDADERO	JAPON	NULL
AR15	CERAMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR16	FERRETERIA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA	NULL
AR18	DEPORTES	PISTOLA OLIMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR19	CONFECCION	BLUSA SRA	101.0566	2000-03-18	VERDADERO	CHINA	NULL
AR20	CERAMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR21	CERAMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPON	NULL
AR23	CONFECCION	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPON	NULL
AR26	JUGUETERIA	FUERTE DE SOLDADOS	143.7020	2000-11-25	VERDADERO	JAPON	NULL
AR27	CONFECCION	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR29	CONFECCION	ABRIGO SRA	360.0736	2001-05-03	VERDADERO	MARRUECOS	NULL
AR30	FERRETERIA	DESTORNILLADOR	9.0584	2002-02-20	VERDADERO	FRANCIA	NULL
AR32	DEPORTES	CRONOMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL

5. Realizar una consulta que muestre los artículos españoles de la sección “Deportes” o aquellos cuyo precio sea superior a 350 € independientemente de cual sea su sección o país de origen.

```
SELECT * FROM PRODUCTOS WHERE (PAISDEORIGEN='ESPAÑA' AND SECCION='DEPORTES') OR PRECIO>350
```

Número de filas: 25

+ Opciones

CÓDIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAISDEORIGEN	FOTO
AR10	JUGUETERIA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA	NULL
AR14	JUGUETERIA	TREN ELECTRICO	1505.3766	2001-07-03	VERDADERO	JAPON	NULL
AR23	CONFECCION	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR27	CONFECCION	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR29	CONFECCION	ABRIGO SRA	360.0736	2001-05-03	VERDADERO	MARRUECOS	NULL
AR32	DEPORTES	CRONOMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

6. Realizar una consulta que muestre los productos cuya fecha esté entre 1/05/2001 y 15/12/2001. En la consulta solo se visualizarán los campos “Nombre de artículo”, “Sección” y “Fecha”.

```
SELECT NOMBREARTICULO, SECCIÓN, FECHA FROM PRODUCTOS WHERE FECHA BETWEEN '2001-05-01' AND '2001-12-15'
```

Número de filas: 25

+ Opciones

NOMBREARTICULO	SECCIÓN	FECHA
LLAVE INGLESA	FERRETERIA	2001-05-23
TREN ELÉCTRICO	JUGUETERIA	2001-07-03
CENICERO	CERÁMICA	2001-07-02
MARTILLO	FERRETERIA	2001-09-04
CAZADORA PIEL	CONFECCIÓN	2001-07-10
BALÓN BALONCESTO	DEPORTES	2001-06-25
ABRIGO SRA	CONFECCIÓN	2001-05-03
PIE DE LÁMPARA	OFICINA	2001-05-27

Capítulo 4.- Cláusula Order By. Ordenador registros

En este capítulo veremos ...

- Cláusula Order BY

ORDER BY: Ordenar los registros seleccionados en función de un campo.

Vamos a realizar una consulta que ya hicimos con anterioridad queremos de la tabla producto los artículos de la sección de DEPORTES o la sección de CERAMICA.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA'
```

Le damos al botón continuar.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

El orden que muestra es el que tiene por defecto es por CÓDIGOARTÍCULO es el orden por el que se introdujo los registros.

Imagínate que los queremos ordenar por sección, para que primero se mostraran todos los registros de CERÁMICA y a continuación los de DEPORTES.

Si es un campo de texto lo ordenará de la A a la Z, si fuera un campo numérico lo ordenará de menor a mayor.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN
```

Le damos a continuar.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL

AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

Ahora observarás que está ordenado por sección.

Por defecto con ORDER BY, la ordenación es en modo creciente de la A a la Z y en números de menor a mayor.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN DESC
```

Añadiendo al final la palabra DESC de descendente se ordenará de la Z a la A y en números de mayor a menor.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL

Ahora primero salen los de DEPORTES y a continuación los de CERÁMICA.

Vamos a ordenarlo por el campo PRECIO.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY PRECIO
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO ¹	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL

No se pueden ordenar por dos campos a la vez, pero si se puede establecer varios criterios de ordenación, varios campos de ordenación.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN
```

Volvemos a ordenar por SECCIÓN.

CÓDIGOARTÍCULO	SECCIÓN ¹	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

Ahora vemos como están ordenados por SECCIÓN, dentro de los artículos de CERÁMICA los ha ordenado, pues no vemos ningún criterio claro, la pregunta es una ver ordenado por SECCIONES, los podemos ordenar por otro campo. Por ejemplo ordenarlos por PRECIO.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN, PRECIO
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN ▲ 1	NOMBREARTÍCULO	PRECIO ▲ 2	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL

Si queremos ordenar SECCIÓN ascendente y PRECIO descendente.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN, PRECIO DESC
```

Este será el resultado.

CÓDIGOARTÍCULO	SECCIÓN ▲ 1	NOMBREARTÍCULO	PRECIO ▼ 2	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

Puedes establecer, 3, 4 y 5 criterios de ordenación si así lo necesitas.

Ahora queremos ordenar por SECCIÓN y PAÍSDEORIGEN.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN, PAÍSDEORIGEN
```

CÓDIGOARTÍCULO	SECCIÓN ▲ 1	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN ▲ 2	FOTO
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL

Ahora si queremos ordenar por SECCIÓN dentro de SECCIÓN ordenarlos por PAÍSDEORIGEN y dentro de PAÍSDEORIGEN por PRECIO.

```
1 | SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' OR SECCIÓN = 'CERAMICA' ORDER BY SECCIÓN, PAÍSDEORIGEN, PRECIO
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN ▲ 1	NOMBREARTÍCULO	PRECIO ▲ 3	FECHA	IMPORTADO	PAÍSDEORIGEN ▲ 2	FOTO
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	VERDADERO	JAPÓN	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL

EJERCICIOS SQL

1. Realizar una consulta que muestre aquellos productos españoles cuya fecha esté comprendida entre febrero y mayo del 2002. Ordenar los resultados por el campo "Nombre artículo" descendientemente.
2. Realizar una consulta que muestre los campos "Empresa", "Dirección" y "Población" de la tabla "Clientes". Ordenar la consulta por el campo "Población" ascendientemente y por "Dirección" descendientemente.
3. Realizar una consulta que muestre todos los campos de la tabla "Clientes". Ordenar los resultados por el campo "Población" ascendientemente.

SOLUCIÓN

1. Realizar una consulta que muestre aquellos productos españoles cuya fecha esté comprendida entre febrero y mayo del 2002. Ordenar los resultados por el campo "Nombre artículo" descendientemente.

```
SELECT * FROM PRODUCTOS WHERE PAÍSDEORIGEN='ESPAÑA' AND FECHA BETWEEN '2002-02-01' AND '2002-05-31' ORDER BY NOMBREARTÍCULO DESC
```

Número de filas: 25 ▼

+ Opciones

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR37	CONFECCIÓN	CINTURÓN DE PIEL	4.3273	2002-05-12	FALSO	ESPAÑA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL

2. Realizar una consulta que muestre los campos “Empresa”, “Dirección” y “Población” de la tabla “Clientes”. Ordenar la consulta por el campo “Población” ascendentemente y por “Dirección” descendentemente.

```
SELECT EMPRESA, DIRECCIÓN, POBLACIÓN FROM CLIENTES ORDER BY POBLACIÓN, DIRECCIÓN DESC
```

1 ▼

Mostrar todo > >>

Número de filas:

25 ▼

+ Opciones

EMPRESA	DIRECCIÓN	POBLACIÓN
CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA
DEPORTES NAUTICOS GARCIA	JUAN FERNANDEZ 89	AVILA
JUGUETES MARTÍNEZ	VIA LAYETANA 245	BARCELONA
JUGUETES EL BARATO	VIA AUGUSTA 245	BARCELONA
JUGUETERIA SUAREZ	PARIS 123	BARCELONA
CONFECCIONES MONICA	MOTORES 67	BARCELONA
EL ESPAÑOLITO	MOTORES 34	BARCELONA
CONFECCIONES RUIZ	LLOBREGAT 345	BARCELONA
EXCLUSIVAS FERNANDEZ	LLOBREGAT 250	BARCELONA
BAZAR EL BARAT	DIAGONAL 56	BARCELONA
ALMACÉN POPULAR	LAS FUENTES 124	BILBAO
CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN
CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN
JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN
DEPORTES MORAN	AUTONOMIA 45	LUGO
FERRETERIA LIMA	VALLECAS 45	MADRID
EXPORTASA	VALLECAS 34	MADRID
FERRETERIA EL CLAVO	PASEO DE ALAMOS 78	MADRID
EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID
JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID
FERRETERIA LA ESCOBA	ORENSE 7	MADRID
BELTRAN E HIJOS	LAS FUENTES 78	MADRID
DEPORTES GARCÍA	GUZMAN EL BUENO 45	MADRID
CONFECCIONES GALAN	FUENCARRAL 78	MADRID
BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID

- Realizar una consulta que muestre todos los campos de la tabla "Clientes". Ordenar los resultados por el campo "Población" ascendentemente.

SELECT * FROM CLIENTES ORDER BY POBLACIÓN

Mostrar todo > >> Número de filas: 25

+ Opciones

CÓDIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACIÓN	TELEFONO	RESPONSABLE	HISTORIAL
CT13	CONFECCIONES ARTIMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL	NULL
CT32	DEPORTES NAUTICOS GARCIA	JUAN FERNANDEZ 85	AVILA	920268648	JUAN CONRADO	NULL
CT27	JUGUETES EL BARATO	VIA AUGUSTA 245	BARCELONA	933486984	ELVIRA IGLESIAS	NULL
CT33	CONFECCIONES RUIZ	LLOBREGAT 345	BARCELONA	934587615	CARLOS SANZ	NULL
CT22	BAZAR EL BARAT	DIAGONAL 56	BARCELONA	936692866	ELISA DAPENA	NULL
CT38	CONFECCIONES MONICA	MOTORES 67	BARCELONA	935681245	PEDRO SERRANO	NULL
CT15	EXCLUSIVAS FERNANDEZ	LLOBREGAT 250	BARCELONA	939558365	LUISA FERNANDEZ	NULL
CT11	JUGUETES MARTINEZ	VIA LAYETANA 245	BARCELONA	936628554	FRANCISCO CUEVAS	NULL
CT08	JUGUETERIA SUAREZ	PARIS 123	BARCELONA	933457866	JUAN GARCIA	NULL
CT03	EL ESPANOLITO	MOTORES 34	BARCELONA	934555343	ANA FERNANDEZ	NULL
CT09	ALMACEN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSE ALVAREZ	NULL
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUIS ALVAREZ	NULL
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZALEZ	NULL
CT18	JUGUETES LA SONRISA	LA BANEZA 67	LEÓN	987945368	FAUSTINO PEREZ	NULL
CT16	DEPORTES MORAN	AUTONOMIA 45	LUGO	982986944	JOSE MANZANO	NULL
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GOMEZ	NULL
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	927785235	LAURA CARRASCO	NULL
CT26	FERRERERIA LA ESCOBA	ORENSE 7	MADRID	918459346	JOSE GARCIA	NULL
CT10	FERRETERIA EL CLAVO	PASEO DE ALAMOS 78	MADRID	914354866	MANUEL MENENDEZ	NULL
CT19	CONFECCIONES GALAN	FUENCARRAL 78	MADRID	913859234	JUAN GARCIA	NULL
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	912495973	ADRIAN ALVAREZ	NULL
CT07	LA CASA DEL JUGUETE	AMERICA 45	MADRID	912649987	ELIAS PEREZ	NULL
CT14	DEPORTES GARCIA	GUZMAN EL BUENO 45	MADRID	913299475	ANA JIMENEZ	NULL
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	915483627	ANGEL SANTAMARIA	NULL
CT01	BELTRAN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTINEZ	NULL

Capítulo 5.- Consulta de agrupación o totales

En este capítulo veremos...

- Consultas de agrupaciones o totales
 - Funciones de agregado

En estas consultas se hacen cálculos por grupos, es decir si cogemos los registros de una tabla los agrupamos en base a un criterio o campo y una vez que los tenemos agrupados con estos registros realizamos un cálculo, puede ser un conteo para saber cuántos artículos tengo de cada sección, puede ser una media para saber cual es el precio medio de los artículos por sección también en la tabla clientes, cuantos clientes tengo de Barcelona, cuantos clientes de Madrid o cuantos artículos tengo de Méjico, artículos americanos, etc.

En los tipos de consultas a las que hay que hacer una agrupación y un cálculo las hacemos a través de las consultas de agrupación o total.

Para poder realizar los cálculos pertinentes si es un conteo, como si es una suma o es una media utilizamos las instrucciones de agregado SQL, cales son estas funciones de agregado.

Funciones de agregado

Función	Descripción
AVG	Calcula el promedio de un campo
COUNT	Cuenta los registros de un campo
SUM	Suma los valores de un campo
MAX	Devuelve el máximo de un campo
MIN	Devuelve el mínimo de un campo

Vamos a comenzar por la tabla de PRODUCTOS.

La primera consulta será cuanto suman los artículos de las SECCIONES.

Nos harán falta dos campos uno que es el de agrupación y el otro que es el del cálculo.

```
1 SELECT SECCIÓN, SUM(PRECIO) FROM PRODUCTOS GROUP BY SECCIÓN
```

Este será el resultado:

SECCIÓN	SUM(PRECIO)
CERÁMICA	442.3498
CONFECCIÓN	501544.2935
DEPORTES	1305.8123
FERRETERÍA	95.3806
JUGUETERÍA	2516.7141
OFICINA	39.7606

Si queremos que salga ordenado por el precio de menor a mayor.

```
1 SELECT SECCIÓN, SUM(PRECIO) FROM PRODUCTOS GROUP BY SECCIÓN ORDER BY PRECIO
```

Vamos a ver el resultado:

SECCIÓN	SUM(PRECIO)	
FERRETERÍA	95.3806	2
OFICINA	39.7606	1
DEPORTES	1305.8123	4
JUGUETERÍA	2516.7141	5
CERÁMICA	442.3498	3
CONFECCIÓN	501544.2935	6

Si os dais cuenta los precios no aparecen ordenados.

Esto es debido a que el campo PRECIO, por si solo no está en la consulta, sino hemos puesto SUM(PRECIO).

Para poder hacerlo nos permite utilizar un ALIAS, consiste en añadir AS y el nuevo nombre para poder identificar esta operación.

```
1 SELECT SECCIÓN, SUM(PRECIO) AS SUMA_ARTICULOS FROM PRODUCTOS GROUP BY SECCIÓN ORDER BY SUMA_ARTICULOS
```

Este será el resultado:

SECCIÓN	SUMA_ARTICULOS	▲ 1
OFICINA	39.7606	
FERRETERÍA	95.3806	
CERÁMICA	442.3498	
DEPORTES	1305.8123	
JUGUETERÍA	2516.7141	
CONFECCIÓN	501544.2935	

Que pasa si agregamos un tercer campo.

```
1 SELECT SECCIÓN, NOMBREARTÍCULO, SUM(PRECIO) AS SUMA_ARTICULO FROM PRODUCTOS  
2 GROUP BY SECCIÓN ORDER BY SUMA_ARTICULO
```

Este será el resultado:

SECCIÓN	NOMBREARTÍCULO	SUMA_ARTICULO ▲ 1
OFICINA	PIE DE LÁMPARA	39.7606
FERRETERÍA	DESTORNILLADOR	95.3806
CERÁMICA	TUBOS	442.3498
DEPORTES	RAQUETA TENIS	1305.8123
JUGUETERÍA	COCHE TELEDIRIGIDO	2516.7141
CONFECCIÓN	TRAJE CABALLERO	501544.2935

Solo aparece el primer artículo de cada sección, en este caso el artículo de menor precio de cada sección.

Esta información es poco relevante. En estas consultas solo dos campos el de agrupación y el de calcular.

Se puede utilizar más de un GROUP BY en una consulta y la tabla nos lo permite.

Otra consulta seria la media de los artículos de las secciones de DEPORTES Y CERÁMICA.

```
1 SELECT SECCIÓN, AVG(PRECIO) FROM PRODUCTOS
2 WHERE SECCIÓN = "DEPORTES" OR SECCIÓN = "CERÁMICA"
3 GROUP BY SECCIÓN
```

Este sería el resultado:

SECCIÓN	AVG(PRECIO)
CERÁMICA	73.72496667
DEPORTES	130.58123000

Otra forma de realizar la consulta y obtener el mismo resultado:

```
1 SELECT SECCIÓN, AVG(PRECIO) FROM PRODUCTOS
2 GROUP BY SECCIÓN HAVING SECCIÓN = "DEPORTES" OR SECCIÓN = "CERÁMICA"
```

SECCIÓN	AVG(PRECIO)
CERÁMICA	73.72496667
DEPORTES	130.58123000

Cuando se utiliza el GROUP BY podemos utilizar el HAVING en lugar del WHERE.

```
1 SELECT SECCIÓN, AVG(PRECIO) AS MEDIA_ART FROM PRODUCTOS
2 GROUP BY SECCIÓN HAVING SECCIÓN = "DEPORTES" OR SECCIÓN = "CERÁMICA"
3 ORDER BY MEDIA_ART DESC
```

Además lo queremos ordenado por la media en modo descendente.

SECCIÓN	MEDIA_ART ▾ 1
DEPORTES	130.58123000
CERÁMICA	73.72496667

Vamos a realizar una consulta que me muestre cuantos clientes tengo de cada provincia.

Para ello vamos a utilizar la función COUNT().

Hay que tener en cuenta que si utilizamos la función COUNT, sobre un campo que en algún registro puede estar vacío, este registro no lo contaría, lo que se suele hacer es utilizar el campo clave como puede ser el CÓDIGOCLIENTE que es un campo que no puede estar vacío.

```
1 SELECT POBLACIÓN, COUNT(CÓDIGOCLIENTE) AS N_CLIENTES FROM CLIENTES
2 GROUP BY POBLACIÓN
```

Este será el resultado:

POBLACIÓN	N_CLIENTES
A CORUÑA	1
ÁVILA	1
BARCELONA	8
BILBAO	1
GIJÓN	2
LEÓN	1
LUGO	1
MADRID	12
MÁLAGA	1
OVIEDO	1
SALAMANCA	2
SANTANDER	1
SEVILLA	2
TERUEL	1
ZAMORA	1
ZARAGOZA	2

Queremos saber el precio del artículo más caro de la sección CONFECCIÓN.

```
1 SELECT SECCIÓN, MAX(PRECIO) AS PRECIO_MAS_ALTO FROM PRODUCTOS
2 GROUP BY SECCIÓN HAVING SECCIÓN = "CONFECCIÓN"
```

Este será el resultado:

SECCIÓN	PRECIO_MAS_ALTO
CONFECCIÓN	500000.0000

Otra forma de realizar esta consulta y obteniendo la misma información.

```
1 SELECT SECCIÓN, MAX(PRECIO) AS PRECIO_MAS_ALTO FROM PRODUCTOS
2 WHERE SECCIÓN = "CONFECCIÓN" GROUP BY SECCIÓN
```

Lo único que tienes que tener en cuenta es el orden.

Este será el resultado:

SECCIÓN	PRECIO_MAS_ALTO
CONFECCIÓN	500000.0000

Queremos incluir el nombre del artículo.

```
1 SELECT SECCIÓN, NOMBREARTÍCULO, MAX(PRECIO) AS PRECIO_MAS_ALTO FROM PRODUCTOS
2 WHERE SECCIÓN = "CONFECCIÓN" GROUP BY SECCIÓN
```

Este será el resultado:

SECCIÓN	NOMBREARTÍCULO	PRECIO_MAS_ALTO
CONFECCIÓN	TRAJE CABALLERO	500000.0000

Si incluimos un tercer campo puede mostrar un NOMBREARTÍCULO que en este caso sea el artículo con el precio más alto, en este caso no ha funcionado.

EJERCICIOS SQL

1. Realizar una consulta sobre la tabla "Clientes" que muestre los campos "Dirección", "Teléfono" y "Población". Este último debe aparecer en la consulta con el nombre de "Residencia". Los registros aparecerán ordenados descendientemente por el campo "Población".
2. Realizar una consulta que muestre que poblaciones hay en la tabla "Clientes".
3. Realizar una consulta de agrupación que muestre la media del precio de los artículos de todas las secciones. Mostrar en la consulta los campos sección y suma por sección.
4. Realizar una consulta de agrupación que muestre la media del precio de todas las secciones menos de juguetería. En la consulta deberán aparecer los campos "Sección" y "Media por sección".
5. Realizar Una consulta que muestre cuantos artículos hay de la sección "Deportes".

|

SOLUCIÓN

1. Realizar una consulta sobre la tabla “Clientes” que muestre los campos “Dirección”, “Teléfono” y “Población”. Este último debe aparecer en la consulta con el nombre de “Residencia”. Los registros aparecerán ordenados descendientemente por el campo “Población”.

```
SELECT DIRECCIÓN, TELÉFONO, POBLACIÓN AS RESIDENCIA FROM CLIENTES ORDER BY RESIDENCIA DESC
```

1 ▼ Mostrar todo > >> | Número de filas: 25 ▼

+ Opciones

DIRECCIÓN	TELÉFONO	RESIDENCIA
EL COMENDADOR 67	975694035	ZARAGOZA
CASTILLA 345	976388934	ZARAGOZA
CARMEN 45	980495288	ZAMORA
ANTON 67	978564025	TERUEL
SIERPES 78	953452567	SEVILLA
INFANTA MERCEDES 23	954869234	SEVILLA
PASEO DEL MAR 45	942049586	SANTANDER
ABEL MARTINEZ 67	921548762	SALAMANCA
CASTELLANOS 205	923548965	SALAMANCA
LA PALOMA 123	985323434	OVIEDO
OLIVARES 3	953756259	MÁLAGA
ORENSE 7	918459346	MADRID
FUENCARRAL 78	913859234	MADRID
ORTEGA Y GASSET 129	927785235	MADRID
ORENSE 89	916872354	MADRID
CASTILLA Y LEÓN 34	915483627	MADRID
VALLECAS 34	913452378	MADRID
AMERICA 45	912649987	MADRID
VALLECAS 45	913532785	MADRID
PASEO DE ALAMOS 78	914354866	MADRID
ATOCHA 55	912495973	MADRID
GUZMAN EL BUENO 45	913299475	MADRID
LAS FUENTES 78	914456435	MADRID
AUTONOMIA 45	982986944	LUGO
LA BAÑEZA 67	987945368	LEÓN

2. Realizar una consulta que muestre que poblaciones hay en la tabla “Clientes”.

```
SELECT POBLACIÓN FROM CLIENTES GROUP BY POBLACIÓN
```

Número de filas: 25 ▼

+ Opciones

POBLACIÓN
A CORUÑA
ÁVILA
BARCELONA
BILBAO
GIJÓN
LEÓN
LUGO
MADRID
MÁLAGA
OVIEDO
SALAMANCA
SANTANDER
SEVILLA
TERUEL
ZAMORA
ZARAGOZA

3. Realizar una consulta de agrupación que muestre la media del precio de los artículos de todas las secciones. Mostrar en la consulta los campos sección y media por sección.

```
SELECT SECCIÓN, SUM(PRECIO) AS MEDIA_POR_SECCIÓN FROM PRODUCTOS GROUP BY SECCIÓN
```

Número de filas: 25 ▼

+ Opciones

SECCIÓN	MEDIA_POR_SECCIÓN
CERÁMICA	442.3498
CONFECCIÓN	501544.2935
DEPORTES	1305.8123
FERRETERÍA	95.3806
JUGUETERÍA	2516.7141
OFICINA	39.7606

4. Realizar una consulta de agrupación que muestre la media del precio de todas las secciones menos de juguetería. En la consulta deberán aparecer los campos “Sección” y “Media por sección”.

```
SELECT SECCIÓN, AVG(PRECIO) AS MEDIA_POR_SECCIÓN FROM PRODUCTOS WHERE SECCIÓN<>'JUGUETERÍA' GROUP BY SECCIÓN
```

Número de filas: 25 ▼

+ Opciones

SECCIÓN	MEDIA_POR_SECCIÓN
CERAMICA	73.72496667
CONFECCION	55727.14372222
DEPORTES	130.58123000
FERRETERÍA	13.62580000
OFICINA	39.76060000

5. Realizar Una consulta que muestre cuantos artículos hay de la sección “Deportes”.

```
SELECT COUNT(SECCIÓN) FROM PRODUCTOS WHERE SECCIÓN='DEPORTES' GROUP BY SECCIÓN
```

+ Opciones

COUNT(SECCIÓN)
10

Capítulo 6.- Consultas de cálculo

En este capítulo veremos...

- Consultas de cálculo
 - Funciones frecuentes:
 - Now()
 - Datediff()
 - Date_format()
 - Round()
 - Truncate
 - Concate()

La función Now(): Nos devuelve el día y la hora actual, en el momento de hacer la consulta.

La función Datediff(): Nos devuelve la diferencia que hay entre dos fechas.

La función Date_format(): Nos permite formatear los resultados.

La función Round(): Nos permite quitar decimales, para redondear, etc.

La función Truncate():

La función Concate(): Sirve para concatenar cadenas de texto.

Vamos a ver algunos ejemplos:

En la tabla PRODUCTOS tenemos un campo llamado PRECIO, pues supongamos que este precio es sin el IVA que es un 21% sobre el valor del producto.

Pues en esta consulta queremos ver el Precio con el IVA cargado.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, PRECIO*1.21 FROM PRODUCTOS
```

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	PRECIO	PRECIO*1.21
DESTORNILLADOR	FERRETERÍA	6.6280	8.019880
TRAJE CABALLERO	CONFECCIÓN	284.5769	344.338049
COCHE TELEDIRIGIDO	JUGUETERÍA	159.4462	192.929902
RAQUETA TENIS	DEPORTES	93.4694	113.097974
MANCUERNAS	DEPORTES	60.0000	72.600000
SERRUCHO	CONFECCIÓN	30.2045	36.547445
CORREPASILLOS	JUGUETERÍA	103.3356	125.036076
PANTALÓN SEÑORA	CONFECCIÓN	174.2310	210.819510
CONSOLA VIDEO	JUGUETERÍA	442.5444	535.478724
TUBOS	CERÁMICA	168.4253	203.794613

LLAVE INGLESA	FERRETERÍA	24.3986	29.522306
CAMISA CABALLERO	CONFECCIÓN	67.1306	81.228026
TREN ELÉCTRICO	JUGUETERÍA	1505.3766	1821.505686
PLATO DECORATIVO	CERÁMICA	54.0911	65.450231
ALICATES	FERRETERÍA	6.7362	8.150802
MUÑECA ANDADORA	JUGUETERÍA	105.0593	127.121753
PISTOLA OLÍMPICA	DEPORTES	46.7347	56.548987
BLUSA SRA.	CONFECCIÓN	101.0566	122.278486
JUEGO DE TE	CERÁMICA	43.2728	52.360088
CENICERO	CERÁMICA	19.7468	23.893628
MARTILLO	FERRETERÍA	11.3952	13.788192
CAZADORA PIEL	CONFECCIÓN	522.6930	632.458530
BALÓN RUGBY	DEPORTES	111.6440	135.089240
BALÓN BALONCESTO	DEPORTES	75.2731	91.080451
FUERTE DE SOLDADOS	JUGUETERÍA	143.7020	173.879420

El nombre de campo, refleja el cálculo, lo más acertado sería PRECIO_CON_IVA, por ejemplo.

Los precios salen con muchos decimales.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, PRECIO*1.21 AS PRECIO_CON_IVA
2 FROM PRODUCTOS
```

Al agregar un alias, este será el resultado:

NOMBREARTÍCULO	SECCIÓN	PRECIO	PRECIO_CON_IVA
DESTORNILLADOR	FERRETERÍA	6.6280	8.019880
TRAJE CABALLERO	CONFECCIÓN	284.5769	344.338049
COCHE TELEDIRIGIDO	JUGUETERÍA	159.4462	192.929902
RAQUETA TENIS	DEPORTES	93.4694	113.097974
MANCUERNAS	DEPORTES	60.0000	72.600000
SERRUCHO	CONFECCIÓN	30.2045	36.547445
CORREPASILLOS	JUGUETERÍA	103.3356	125.036076

Con respecto a los decimales vamos a utilizar la función Round()

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, ROUND(PRECIO*1.21, 2) AS PRECIO_CON_IVA
2 FROM PRODUCTOS
```

ROUND('Cálculo', número de decimales) → Tal como se muestra en el ejemplo anterior.

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	PRECIO	PRECIO_CON_IVA
DESTORNILLADOR	FERRETERÍA	6.6280	8.02
TRAJE CABALLERO	CONFECCIÓN	284.5769	344.34
COCHE TELEDIRIGIDO	JUGUETERÍA	159.4462	192.93
RAQUETA TENIS	DEPORTES	93.4694	113.10
MANCUERNAS	DEPORTES	60.0000	72.60
SERRUCHO	CONFECCIÓN	30.2045	36.55
CORREPASILLOS	JUGUETERÍA	103.3356	125.04

Ahora queremos crear una consulta la cual refleje un descuento de 3€ a cada uno de los artículos, después de haberle incrementado el IVA.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, ROUND(PRECIO*1.21, 2) AS PRECIO_CON_IVA,
2 ROUND(PRECIO*1.21 -3, 2) AS PRECIO_DESCUENTO FROM PRODUCTOS
```

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	PRECIO	PRECIO_CON_IVA	PRECIO_DESCUENTO
DESTORNILLADOR	FERRETERÍA	6.6280	8.02	5.02
TRAJE CABALLERO	CONFECCIÓN	284.5769	344.34	341.34
COCHE TELEDIRIGIDO	JUGUETERÍA	159.4462	192.93	189.93
RAQUETA TENIS	DEPORTES	93.4694	113.10	110.10
MANCUERNAS	DEPORTES	60.0000	72.60	69.60
SERRUCHO	CONFECCIÓN	30.2045	36.55	33.55
CORREPASILLOS	JUGUETERÍA	103.3356	125.04	122.04

Ahora queremos consultar de la tabla PRODUCTOS los campos NOMBREARTÍCULO, SECCIÓN Y FECHA, donde la SECCIÓN sea igual a "DEPORTES".

Vamos a ver el resultado:

NOMBREARTÍCULO	SECCIÓN	FECHA
RAQUETA TENIS	DEPORTES	2000-03-20
MANCUERNAS	DEPORTES	2000-09-13
PISTOLA OLÍMPICA	DEPORTES	2001-02-02
BALÓN RUGBY	DEPORTES	2000-11-11
BALÓN BALONCESTO	DEPORTES	2001-06-25
BALÓN FÚTBOL	DEPORTES	2002-07-04
CRONÓMETRO	DEPORTES	2002-01-03

CAÑA DE PESCA	DEPORTES	2000-02-14
BOTA ALPINISMO	DEPORTES	2002-05-05
PALAS DE PING PONG	DEPORTES	2002-02-02

Queremos realizar un cálculo entre fechas, para ello utilizaremos la función NOW() que nos retorna la fecha y la hora actual.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, FECHA, NOW() AS DIA_DE_HOY FROM PRODUCTOS
2 WHERE SECCIÓN = "DEPORTES"
```

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	FECHA	DIA_DE_HOY
RAQUETA TENIS	DEPORTES	2000-03-20	2023-03-03 11:00:44
MANCUERNAS	DEPORTES	2000-09-13	2023-03-03 11:00:44
PISTOLA OLÍMPICA	DEPORTES	2001-02-02	2023-03-03 11:00:44
BALÓN RUGBY	DEPORTES	2000-11-11	2023-03-03 11:00:44
BALÓN BALONCESTO	DEPORTES	2001-06-25	2023-03-03 11:00:44
BALÓN FÚTBOL	DEPORTES	2002-07-04	2023-03-03 11:00:44
CRONÓMETRO	DEPORTES	2002-01-03	2023-03-03 11:00:44
CAÑA DE PESCA	DEPORTES	2000-02-14	2023-03-03 11:00:44
BOTA ALPINISMO	DEPORTES	2002-05-05	2023-03-03 11:00:44
PALAS DE PING PONG	DEPORTES	2002-02-02	2023-03-03 11:00:44

Ahora podemos agregar otro campo que nos diga la diferencia en días que hay entre las dos fechas con la función DATEDIFF() nos calcula los días de diferencia entre las dos fechas.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, FECHA, NOW() AS DIA_DE_HOY, DATEDIFF(NOW(),FECHA) AS DÍAS
2 FROM PRODUCTOS WHERE SECCIÓN = "DEPORTES"
```

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	FECHA	DIA_DE_HOY	DÍAS
RAQUETA TENIS	DEPORTES	2000-03-20	2023-03-03 11:19:02	8383
MANCUERNAS	DEPORTES	2000-09-13	2023-03-03 11:19:02	8206
PISTOLA OLÍMPICA	DEPORTES	2001-02-02	2023-03-03 11:19:02	8064
BALÓN RUGBY	DEPORTES	2000-11-11	2023-03-03 11:19:02	8147
BALÓN BALONCESTO	DEPORTES	2001-06-25	2023-03-03 11:19:02	7921
BALÓN FÚTBOL	DEPORTES	2002-07-04	2023-03-03 11:19:02	7547
CRONÓMETRO	DEPORTES	2002-01-03	2023-03-03 11:19:02	7729

CAÑA DE PESCA	DEPORTES	2000-02-14	2023-03-03 11:19:02	8418
BOTA ALPINISMO	DEPORTES	2002-05-05	2023-03-03 11:19:02	7607
PALAS DE PING PONG	DEPORTES	2002-02-02	2023-03-03 11:19:02	7699

Ahora si observáis la fecha de hoy aparece la fecha en formato año, mes y día, seguido de la hora, minutos y segundos.

Tenemos una función llamada DATE_FORMAT que nos permite modificar el formato de un campo fecha, hay distintos formatos:

('%D+%M') → Solo saldrá el día y el mes.

('%D+%M+%Y') → Muestra el día, el mes y el año.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN, FECHA, DATE_FORMAT(NOW(), '%D-%M-%Y') AS DIA_DE_HOY,
2 DATEDIFF(NOW(), FECHA) AS DÍAS FROM PRODUCTOS WHERE SECCIÓN = "DEPORTES"
```

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN	FECHA	DIA_DE_HOY	DÍAS
RAQUETA TENIS	DEPORTES	2000-03-20	3rd-March-2023	8383
MANCUERNAS	DEPORTES	2000-09-13	3rd-March-2023	8206
PISTOLA OLÍMPICA	DEPORTES	2001-02-02	3rd-March-2023	8064
BALÓN RUGBY	DEPORTES	2000-11-11	3rd-March-2023	8147
BALÓN BALONCESTO	DEPORTES	2001-06-25	3rd-March-2023	7921
BALÓN FÚTBOL	DEPORTES	2002-07-04	3rd-March-2023	7547
CRONÓMETRO	DEPORTES	2002-01-03	3rd-March-2023	7729
CAÑA DE PESCA	DEPORTES	2000-02-14	3rd-March-2023	8418
BOTA ALPINISMO	DEPORTES	2002-05-05	3rd-March-2023	7607
PALAS DE PING PONG	DEPORTES	2002-02-02	3rd-March-2023	7699

EJERCICIOS SQL

1. Realizar una consulta que visualice los campos **NOMBRE ARTÍCULO**, **SECCIÓN**, **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto "**DESCUENTO_7**". Debe mostrar el resultado de aplicar sobre el campo **PRECIO** un descuento de un 7 %. El formato del nuevo campo para debe aparecer con 2 lugares decimales.
2. Realizar una consulta visualizando los campos **FECHA**, **SECCIÓN**, **NOMBRE ARTÍCULO** y **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto "**DTO2 €_EN_CERÁMICA**". Debe mostrar el resultado de aplicar sobre el campo **PRECIO** la resta de 2 € sólo a los artículos de la sección **CERÁMICA**. El formato del nuevo campo debe aparecer con 2 lugares decimales. Ordenar el resultado de la consulta por el campo **FECHA** descendente.
3. Realizar una consulta visualizando los campos **NOMBRE ARTÍCULO**, **SECCIÓN**, **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto "**PRECIO_AUMENTADO_EN_2**". Debe mostrar el **PRECIO** con un incremento de un 2% del **PRECIO**. Sólo debemos tener en cuenta los registros de la sección **FERRETERÍA**. El nuevo campo debe aparecer en Euros y con 2 lugares decimales.

SOLUCIÓN

1. Realizar una consulta que visualice los campos **NOMBRE ARTÍCULO**, **SECCIÓN**, **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto "**DESCUENTO_7**". Debe mostrar el resultado de aplicar sobre el campo **PRECIO** un descuento de un 7 %. El formato del nuevo campo para debe aparecer con 2 lugares decimales.

✓ Mostrando filas 0 - 24 (total de 46, La consulta tardó 0.0000 seg)

```
SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, ROUND(PRECIO-(PRECIO*0.07),2) AS DESCUENTO_7 FROM PRODUCTOS
```

1 ▼ Mostrar todo > >> | Número de filas: 25 ▼

+ Opciones

NOMBREARTÍCULO	SECCIÓN	PRECIO	DESCUENTO_7
DESTORNILLADOR	FERRETERIA	6.6280	6.16
TRAJE CABALLERO	CONFECCION	284.5769	264.66
COCHE TELEDIRIGIDO	JUGUETERIA	159.4462	148.28
RAQUETA TENIS	DEPORTES	93.4694	86.93
MANCUERNAS	DEPORTES	60.0000	55.80
SERRUCHO	CONFECCIÓN	30.2045	28.09
CORREPASILLOS	JUGUETERIA	103.3356	96.10
PANTALON SEÑORA	CONFECCION	174.2310	162.03
CONSOLA VIDEO	JUGUETERIA	442.5444	411.57
TUBOS	CERAMICA	168.4253	156.64
LLAVE INGLESA	FERRETERIA	24.3986	22.69
CAMISA CABALLERO	CONFECCION	67.1306	62.43
TREN ELECTRICO	JUGUETERIA	1505.3766	1400.00
PLATO DECORATIVO	CERAMICA	54.0911	50.30
ALICATES	FERRETERIA	6.7362	6.26
MUÑECA ANDADORA	JUGUETERIA	105.0593	97.71

- Realizar una consulta visualizando los campos **FECHA**, **SECCIÓN**, **NOMBRE ARTÍCULO** y **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto “**DTO2€_EN_CERÁMICA**”. Debe mostrar el resultado de aplicar sobre el campo **PRECIO** la resta de 2 € sólo a los artículos de la sección **CERÁMICA**. El formato del nuevo campo debe aparecer con 2 lugares decimales. Ordenar el resultado de la consulta por el campo **FECHA** descendente.

Mostrando filas 0 - 5 (total de 6, La consulta tardó 0.0560 seg) [FECHA: 2002-09-02 - 2000-02-04]

```
SELECT FECHA, SECCIÓN, NOMBREARTÍCULO, PRECIO, ROUND(PRECIO-2,2) AS DTO2€_EN_CERAMICA FROM PRODUCTOS WHERE SECCIÓN='CERÁMICA' ORDER BY FECHA DESC
```

Número de filas: 25

+ Opciones

FECHA	SECCION	NOMBREARTICULO	PRECIO	DTO2€_EN_CERAMICA
2002-09-02	CERÁMICA	JARRA CHINA	127.7704	125.77
2001-07-02	CERÁMICA	CENICERO	19.7468	17.75
2001-01-15	CERÁMICA	JUEGO DE TE	43.2728	41.27
2000-06-07	CERÁMICA	PLATO DECORATIVO	54.0911	52.09
2000-02-23	CERÁMICA	MACETA	29.0434	27.04
2000-02-04	CERÁMICA	TUBOS	168.4253	166.43

- Realizar una consulta visualizando los campos **NOMBRE ARTÍCULO**, **SECCIÓN**, **PRECIO** de la tabla **PRODUCTOS** y un campo nuevo que nombramos con el texto “**PRECIO_AUMENTADO_EN_2**”. Debe mostrar el **PRECIO** con un incremento de un 2% del **PRECIO**. Sólo debemos tener en cuenta los registros de la sección **FERRETERÍA**. El nuevo campo debe aparecer en Euros y con 2 lugares decimales.

Mostrando filas 0 - 6 (total de 7, La consulta tardó 0.0000 seg)

```
SELECT NOMBREARTÍCULO, SECCIÓN, PRECIO, ROUND(PRECIO*1.02,2) AS PRECIO_AUMENTADO_EN_2 FROM PRODUCTOS WHERE SECCIÓN='FERRETERÍA'
```

Número de filas: 25

+ Opciones

NOMBREARTICULO	SECCION	PRECIO	PRECIO_AUMENTADO_EN_2
DESTORNILLADOR	FERRETERIA	6.6280	6.76
LLAVE INGLESA	FERRETERIA	24.3986	24.89
ALICATES	FERRETERIA	6.7362	6.87
MARTILLO	FERRETERIA	11.3952	11.62
DESTORNILLADOR	FERRETERIA	9.0584	9.24
LIMA GRANDE	FERRETERIA	22.0692	22.51
JUEGO DE BROCAS	FERRETERIA	15.0950	15.40

Número de filas: 25

Capítulo 7.- Consultas Multitabla I

En este capítulo veremos...

- Consultas Multitabla / Consultas de Unión
 - Unión Externa:
 - Union
 - Union All
 - Except
 - Intersect
 - Minus
 - Unión interna
 - Inner Join
 - Left Join
 - Right Join

Hasta ahora toda las consultas que hemos realizado se basa en una única tabla, pero imaginarnos que en una base de datos en la cual tenemos 30 tablas queremos realizar una consulta en la queremos ver la información de 5 tablas.

Estas consultas de unión se dividen en dos grandes grupos, consultas de Unión Externa y consultas de Unión Interna.

Con las prácticas que realizaremos a continuación iremos viendo las diferencias que hay.

En principio nos vamos a trabajar con las consultas de Unión Externa.

Operador Union:

Consultas con Union

Tabla 1

CÓDIGO ARTÍCULO	SECCIÓN	NOMBRE ARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍS DE ORIGEN
AR01	FERRERÍA	DESTORNILLADOR	6,63 €	22/10/2000	✓	ESPAÑA
AR02	CONFECCIÓN	TRAJE CABALLERO	284,58 €	11/03/2002	✓	ITALIA
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159,45 €	26/05/2002	✓	MARRUECOS
AR04	DEPORTES	RACQUETA TENIS	93,47 €	20/03/2000	✓	USA
AR06	DEPORTES	MANCUERNAS	60,00 €	13/09/2000	✓	USA
AR07	CONFECCIÓN	SERRUCHO	30,20 €	23/03/2001	✓	FRANCIA
AR08	JUGUETERÍA	CORREPAJULLOS	103,34 €	11/04/2000	✓	JAPÓN
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174,23 €	10/01/2000	✓	MARRUECOS
AR10	JUGUETERÍA	CONSOLA VIDEO	442,54 €	24/09/2002	✓	USA
AR11	CERÁMICA	TUBOS	388,81 €	04/03/2000	✓	CHINA
AR12	FERRERÍA	LLAVES INGLESA	24,40 €	23/05/2001	✓	USA
AR13	CONFECCIÓN	CAMISA CABALLERO	67,13 €	11/08/2002	✓	ESPAÑA
AR14	JUGUETERÍA	TREN ELECTRICO	1.505,38 €	03/07/2001	✓	JAPÓN
AR15	CERÁMICA	PLATO DECORATIVO	94,09 €	07/06/2000	✓	CHINA
AR16	FERRERÍA	ALICATES	6,74 €	17/04/2000	✓	ITALIA

Tabla 2

CÓDIGO ART.	SECCIÓN	NOMBRE ARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍS DE ORIG.	FOTO
AR01	ALTA COSTURA	TRAJE CABALLERO	1.284,58 €	11/03/2002	-	ITALIA	
AR01	DEPORTES DE RIESGO	RAQUETA TENIS	1.093,47 €	20/03/2000	-	USA	
AR02	DEPORTES DE RIESGO	MANCUERNAS	1.060,00 €	13/09/2000	-	USA	
AR03	ALTA COSTURA	SERRUCHO	1.030,20 €	23/03/2001	-	FRANCIA	
AR04	ALTA COSTURA	PANTALÓN SEÑORA	1.174,23 €	10/01/2000	-	MARRUECOS	
AR05	ALTA COSTURA	CAMISA CABALLERO	1.067,13 €	11/08/2002	0	ESPAÑA	
AR06	DEPORTES DE RIESGO	PISTOLA OLÍMPICA	1.046,73 €	02/02/2001	-	SUECIA	
AR07	ALTA COSTURA	BUSA SRA.	1.102,06 €	18/03/2000	-	CHINA	
AR08	ALTA COSTURA	CAZADORA PIEL	1.522,69 €	10/07/2001	-	ITALIA	
AR09	DEPORTES DE RIESGO	BALÓN RUGBY	1.111,64 €	11/11/2000	-	USA	
AR00	DEPORTES DE RIESGO	BALÓN BALONCESTO	1.075,27 €	25/06/2001	-	JAPÓN	
AR01	ALTA COSTURA	ABRIGO CABALLERO	1.560,00 €	05/04/2002	-	ITALIA	
AR02	DEPORTES DE RIESGO	BALÓN FÚTBOL	1.043,91 €	04/07/2002	0	ESPAÑA	
AR03	ALTA COSTURA	ABRIGO SRA.	1.360,07 €	03/05/2001	-	MARRUECOS	
AR04	DEPORTES DE RIESGO	CRONÓMETRO	1.435,18 €	03/01/2002	-	USA	

Permite unir en una consulta varias tablas que podemos tener almacenadas en nuestra base de datos.

Cuando uno maneja una base de datos con el tiempo se suele encontrar con una situación bastante frecuente, tener la misma información pero almacenar en una única tabla dividida la información en varias tablas.

Que ocurre cuando tenemos la información dividida en varias tablas, pues cuando haces una consulta iría muy bien unir la información esas dos tablas en una única consulta, para poder ver esta información de golpe.

Para poder utilizar Union, estas tablas deben de cumplir una serie de requisitos:

El primer requisito es que las dos tablas deben de tener el mismo número de campos, además tienen que tener tipos de datos compatibles.

El nombre de los campos pueden ser diferentes.

Lo que ocurre es que las dos tablas quedan como si fueran una sola tabla.

Consultas con Union

CÓDIGO ARTÍCULO	SECCIÓN	NOMBRE ARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍS DE ORIGEN
AR01	FERRETERÍA	DESTORNILLADOR	6,63 €	22/10/2000	✓	ESPAÑA
AR02	CONFECCIÓN	TRAJE CABALLERO	284,58 €	11/03/2002	✓	ITALIA
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159,45 €	26/05/2002	✓	MARRUECOS
AR04	DEPORTES	RAQUETA TENIS	93,47 €	20/03/2000	✓	USA
AR06	DEPORTES	MANCUERNAS	60,00 €	13/09/2000	✓	USA
AR07	CONFECCIÓN	SERRUCHO	30,20 €	23/03/2001	✓	FRANCIA
AR08	JUGUETERÍA	CORREPASILLOS	103,34 €	11/04/2000	✓	JAPÓN
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174,23 €	10/01/2000	✓	MARRUECOS
AR10	JUGUETERÍA	CONSOLA VIDEO	442,54 €	24/09/2002	✓	USA
AR11	CERÁMICA	TUBOS	168,43 €	04/02/2000	✓	CHINA
AR12	FERRETERÍA	LLAVE INGLESA	24,40 €	23/05/2001	✓	USA
AR13	CONFECCIÓN	CAMISA CABALLERO	67,13 €	11/08/2002	✓	ESPAÑA
AR14	JUGUETERÍA	TREN ELÉCTRICO	1.505,38 €	03/07/2001	✓	JAPÓN
AR15	CERÁMICA	PLATO DECORATIVO	54,09 €	07/06/2000	✓	CHINA
AR16	FERRETERÍA	ALICATES	6,74 €	17/04/2000	✓	ITALIA
AR50	ALTA COSTURA	TRAJE CABALLERO	1.284,58 €	11/03/2002	-1	ITALIA
AR51	DEPORTES DE RIESGO	RAQUETA TENIS	1.093,47 €	20/03/2000	-1	USA
AR52	DEPORTES DE RIESGO	MANCUERNAS	1.060,00 €	13/09/2000	-1	USA
AR53	ALTA COSTURA	SERRUCHO	1.030,20 €	23/03/2001	-1	FRANCIA
AR54	ALTA COSTURA	PANTALÓN SEÑORA	1.174,23 €	10/01/2000	-1	MARRUECOS
AR55	ALTA COSTURA	CAMISA CABALLERO	1.067,13 €	11/08/2002	0	ESPAÑA
AR56	DEPORTES DE RIESGO	PISTOLA OLÍMPICA	1.046,73 €	02/02/2001	-1	SUECIA
AR57	ALTA COSTURA	BLUSA SRA.	1.101,06 €	18/03/2000	-1	CHINA
AR58	ALTA COSTURA	CAZADORA PIEL	1.522,69 €	10/07/2001	-1	ITALIA
AR59	DEPORTES DE RIESGO	BALÓN RUGBY	1.111,64 €	11/11/2000	-1	USA
AR60	DEPORTES DE RIESGO	BALÓN BALONCESTO	1.075,27 €	25/06/2001	-1	JAPÓN
AR61	ALTA COSTURA	ABRIGO CABALLERO	1.500,00 €	05/04/2002	-1	ITALIA
AR62	DEPORTES DE RIESGO	BALÓN FÚTBOL	1.043,91 €	04/07/2002	0	ESPAÑA
AR63	ALTA COSTURA	ABRIGO SRA	1.360,07 €	03/05/2001	-1	MARRUECOS
AR64	DEPORTES DE RIESGO	CRONÓMETRO	1.439,18 €	03/01/2002	-1	USA

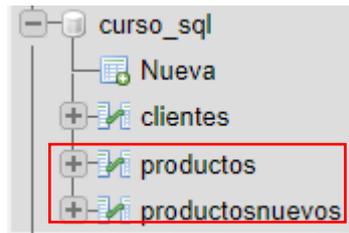
Esta consulta toma como nombres los nombre de la tabla1.

En el siguiente enlace podrás descargar esta segunda tabla para realizar los ejercicios.

Haz clic sobre la hoja de cálculo.



Como hicimos en uno de los primeros capítulos vamos a importar la hoja de cálculo.



Ya tenemos las dos tablas con número de campos idénticos, campos de mismo tipo, etc.

Recuerda que los campos PRECIO y MONEDA tiene uno ser de tipo moneda y el otro de tipo fecha.

En una tabla tenemos la SECCIÓN de DEPORTES y en la segunda tabla DEPORTES DE RIESGO y queremos realizar una consulta para ver las dos secciones.

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = "DEPORTES" UNION
2 SELECT * FROM PRODUCTOSNUEVOS WHERE SECCIÓN = 'DEPORTES DE RIESGO'
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR51	DEPORTES DE RIESGO	RAQUETA TENIS	1.0000	2000-03-20	VERDADERO	USA	NULL
AR52	DEPORTES DE RIESGO	MANCUERNAS	1.0000	2000-09-13	VERDADERO	USA	NULL
AR56	DEPORTES DE RIESGO	PISTOLA OLÍMPICA	1.0000	2001-02-02	VERDADERO	SUECIA	NULL
AR59	DEPORTES DE RIESGO	BALÓN RUGBY	1.0000	2000-11-11	VERDADERO	USA	NULL
AR60	DEPORTES DE RIESGO	BALÓN BALONCESTO	1.0000	2001-06-25	VERDADERO	JAPÓN	NULL
AR62	DEPORTES DE RIESGO	BALÓN FÚTBOL	1.0000	2002-07-04	FALSO	ESPAÑA	NULL
AR64	DEPORTES DE RIESGO	CRONÓMETRO	1.0000	2002-01-03	VERDADERO	USA	NULL
AR66	DEPORTES DE RIESGO	CAÑA DE PESCA	1.0000	2000-02-14	VERDADERO	USA	NULL
AR67	DEPORTES DE RIESGO	BOTA ALPINISMO	1.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR68	DEPORTES DE RIESGO	PALAS DE PING PONG	1.0000	2002-02-02	FALSO	ESPAÑA	NULL

Los nombres de los campos son los nombres de la tabla1.

Ahora vamos a hacer una consulta que nos muestre los artículos de la tabla de PRODUCTOS cuyo precio es superior a 500 € y los artículos de la tabla PRODUCTOSNUEVOS cuya Sección sea ALTA COSTURA.

```
1 SELECT * FROM PRODUCTOS WHERE PRECIO > 500 UNION
2 SELECT * FROM PRODUCTOSNUEVOS WHERE SECCIÓN = 'ALTA COSTURA'
```

El resultado será:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	VERDADERO	JAPÓN	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR50	ALTA COSTURA	TRAJE CABALLERO	1.0000	2002-03-11	VERDADERO	ITALIA	NULL
AR53	ALTA COSTURA	SERRUCHO	1.0000	2001-03-23	VERDADERO	FRANCIA	NULL
AR54	ALTA COSTURA	PANTALÓN SEÑORA	1.0000	2000-01-10	VERDADERO	MARRUECOS	NULL
AR55	ALTA COSTURA	CAMISA CABALLERO	1.0000	2002-08-11	FALSO	ESPAÑA	NULL
AR57	ALTA COSTURA	BLUSA SRA.	1.0000	2000-03-18	VERDADERO	CHINA	NULL
AR58	ALTA COSTURA	CAZADORA PIEL	2.0000	2001-07-10	VERDADERO	ITALIA	NULL
AR61	ALTA COSTURA	ABRIGO CABALLERO	2.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR63	ALTA COSTURA	ABRIGO SRA	1.0000	2001-05-03	VERDADERO	MARRUECOS	NULL
AR65	ALTA COSTURA	CINTURÓN DE PIEL	1.0000	2002-05-12	FALSO	ESPAÑA	NULL

Ahora vamos a trabajar con el operador UNION ALL.

Una diferencia entre UNION y UNION ALL, en el caso de registros repetidos UNION lo muestra solamente una vez mientras que en UNION ALL cuando hay registros repetidos en las tablas de unión Tabla1 y Tabla2, lo muestra siempre que aparezca.

Seleccionando la tabla PRODUCTOSNUEVOS en la pestaña de insertar agregamos un registro idéntico como lo tiene la tabla PRODUCTOS.

Columna	Tipo	Función	Nulo	Valor
CÓDIGOARTÍCULO	varchar(4)	<input type="text"/>	<input type="checkbox"/>	AR06
SECCIÓN	varchar(18)	<input type="text"/>	<input type="checkbox"/>	DEPORTES
NOMBREARTÍCULO	varchar(18)	<input type="text"/>	<input type="checkbox"/>	MANCUERNAS
PRECIO	decimal(8,0)	<input type="text"/>	<input type="checkbox"/>	60.0000
FECHA	date	<input type="text"/>	<input type="checkbox"/>	2000-09-13
IMPORTADO	varchar(9)	<input type="text"/>	<input type="checkbox"/>	VERDADERO
PAÍSDEORIGEN	varchar(9)	<input type="text"/>	<input type="checkbox"/>	USA
FOTO	varchar(10)	<input type="text"/>	<input checked="" type="checkbox"/>	<input type="text"/>

A continuación le damos al botón continuar.

Ahora las dos tablas tienen el mismo registro.

Ahora vamos a realizar la consulta de unión.

```

1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' UNION
2 SELECT * FROM PRODUCTOSNUEVOS |

```

Le damos al botón continuar.

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL
AR50	ALTA COSTURA	TRAJE CABALLERO	1.0000	2002-03-11	VERDADERO	ITALIA	NULL
AR51	DEPORTES DE RIESGO	RAQUETA TENIS	1.0000	2000-03-20	VERDADERO	USA	NULL
AR52	DEPORTES DE RIESGO	MANCUERNAS	1.0000	2000-09-13	VERDADERO	USA	NULL
AR53	ALTA COSTURA	SERRUCHO	1.0000	2001-03-23	VERDADERO	FRANCIA	NULL
AR54	ALTA COSTURA	PANTALÓN SEÑORA	1.0000	2000-01-10	VERDADERO	MARRUECOS	NULL
AR55	ALTA COSTURA	CAMISA CABALLERO	1.0000	2002-08-11	FALSO	ESPAÑA	NULL
AR56	DEPORTES DE RIESGO	PISTOLA OLÍMPICA	1.0000	2001-02-02	VERDADERO	SUECIA	NULL
AR57	ALTA COSTURA	BLUSA SRA.	1.0000	2000-03-18	VERDADERO	CHINA	NULL
AR58	ALTA COSTURA	CAZADORA PIEL	2.0000	2001-07-10	VERDADERO	ITALIA	NULL
AR59	DEPORTES DE RIESGO	BALÓN RUGBY	1.0000	2000-11-11	VERDADERO	USA	NULL
AR60	DEPORTES DE RIESGO	BALÓN BALONCESTO	1.0000	2001-06-25	VERDADERO	JAPÓN	NULL
AR61	ALTA COSTURA	ABRIGO CABALLERO	2.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR62	DEPORTES DE RIESGO	BALÓN FÚTBOL	1.0000	2002-07-04	FALSO	ESPAÑA	NULL
AR63	ALTA COSTURA	ABRIGO SRA	1.0000	2001-05-03	VERDADERO	MARRUECOS	NULL
AR64	DEPORTES DE RIESGO	CRONÓMETRO	1.0000	2002-01-03	VERDADERO	USA	NULL
AR65	ALTA COSTURA	CINTURÓN DE PIEL	1.0000	2002-05-12	FALSO	ESPAÑA	NULL
AR66	DEPORTES DE RIESGO	CAÑA DE PESCA	1.0000	2000-02-14	VERDADERO	USA	NULL
AR67	DEPORTES DE RIESGO	BOTA ALPINISMO	1.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR68	DEPORTES DE RIESGO	PALAS DE PING PONG	1.0000	2002-02-02	FALSO	ESPAÑA	NULL

```
1 SELECT * FROM PRODUCTOS WHERE SECCIÓN = 'DEPORTES' UNION ALL
2 SELECT * FROM PRODUCTOSNUEVOS |
```

Si lo cambiamos por UNION ALL

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

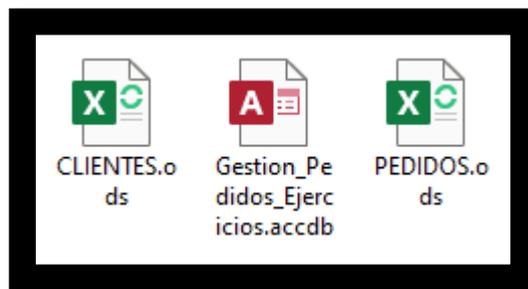
AR50	ALTA COSTURA	TRAJE CABALLERO	1.0000	2002-03-11	VERDADERO	ITALIA	NULL
AR51	DEPORTES DE RIESGO	RAQUETA TENIS	1.0000	2000-03-20	VERDADERO	USA	NULL
AR52	DEPORTES DE RIESGO	MANCUERNAS	1.0000	2000-09-13	VERDADERO	USA	NULL
AR53	ALTA COSTURA	SERRUCHO	1.0000	2001-03-23	VERDADERO	FRANCIA	NULL
AR54	ALTA COSTURA	PANTALÓN SEÑORA	1.0000	2000-01-10	VERDADERO	MARRUECOS	NULL
AR55	ALTA COSTURA	CAMISA CABALLERO	1.0000	2002-08-11	FALSO	ESPAÑA	NULL
AR56	DEPORTES DE RIESGO	PISTOLA OLÍMPICA	1.0000	2001-02-02	VERDADERO	SUECIA	NULL
AR57	ALTA COSTURA	BLUSA SRA.	1.0000	2000-03-18	VERDADERO	CHINA	NULL
AR58	ALTA COSTURA	CAZADORA PIEL	2.0000	2001-07-10	VERDADERO	ITALIA	NULL
AR59	DEPORTES DE RIESGO	BALÓN RUGBY	1.0000	2000-11-11	VERDADERO	USA	NULL
AR60	DEPORTES DE RIESGO	BALÓN BALONCESTO	1.0000	2001-06-25	VERDADERO	JAPÓN	NULL
AR61	ALTA COSTURA	ABRIGO CABALLERO	2.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR62	DEPORTES DE RIESGO	BALÓN FÚTBOL	1.0000	2002-07-04	FALSO	ESPAÑA	NULL
AR63	ALTA COSTURA	ABRIGO SRA	1.0000	2001-05-03	VERDADERO	MARRUECOS	NULL
AR64	DEPORTES DE RIESGO	CRONÓMETRO	1.0000	2002-01-03	VERDADERO	USA	NULL
AR65	ALTA COSTURA	CINTURÓN DE PIEL	1.0000	2002-05-12	FALSO	ESPAÑA	NULL
AR66	DEPORTES DE RIESGO	CAÑA DE PESCA	1.0000	2000-02-14	VERDADERO	USA	NULL
AR67	DEPORTES DE RIESGO	BOTA ALPINISMO	1.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR68	DEPORTES DE RIESGO	PALAS DE PING PONG	1.0000	2002-02-02	FALSO	ESPAÑA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL

En este caso el registro duplicado se muestra.

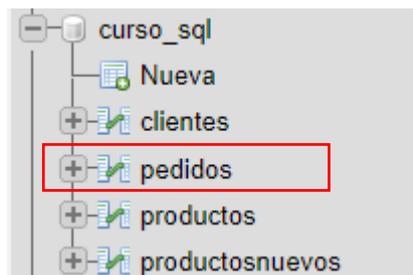
Capítulo 8.- Consultas Multitabla II Inner Join

- Unión interna
 - Inner join
 - Left Join
 - Right Join

Para este capítulo tienes que descargar el siguiente archivo haciendo clic en la siguiente imagen.



Vamos a importar el archivo PEDIDOS.ods, pero antes vamos a ver como están los campos de la tabla.



Ya hemos agregado pedidos.

Si abrimos la tabla CLIENTES nos daremos cuenta que refleja los datos de una serie de clientes.

CÓDIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACIÓN	TELÉFONO	RESPONSABLE	HISTORIAL
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTÍNEZ	NULL
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA	NULL
CT03	EL ESPAÑOLITO	MOTORES 34	BARCELONA	934565343	ANA FERNÁNDEZ	NULL
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GÓMEZ	NULL
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ	NULL
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	912649987	ELÍAS PÉREZ	NULL
CT08	JUGUETERÍA SUÁREZ	PARIS 123	BARCELONA	933457866	JUAN GARCÍA	NULL
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ	NULL
CT10	FERETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	914354866	MANUEL MENÉNDEZ	NULL
CT11	JUGUETES MARTÍNEZ	VIA LAYETANA 245	BARCELONA	936628554	FRANCISCO CUEVAS	NULL
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO	NULL
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL	NULL
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	913299475	ANA JIMÉNEZ	NULL
CT15	EXCLUSIVAS FERNÁNDEZ	LLOBREGAT 250	BARCELONA	939558365	LUISA FERNÁNDEZ	NULL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO	NULL
CT17	BAZAR FRANCISCO	CARMEN 45	ZAMORA	980495288	CARLOS BELTRÁN	NULL
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ	NULL
CT19	CONFECCIONES GALÁN	FUENCARRAL 78	MADRID	913859234	JUAN GARCÍA	NULL

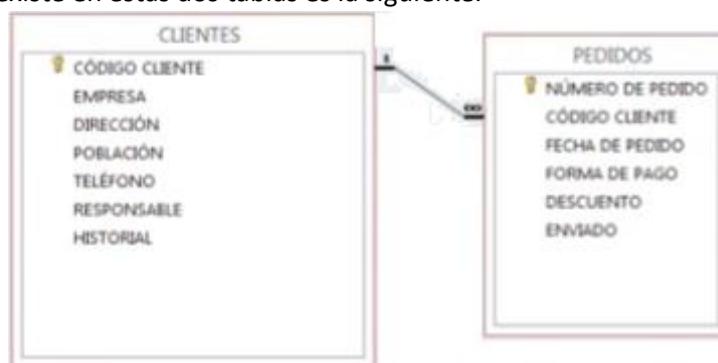
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ	NULL
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO	NULL
CT22	BAZAR EL BARAT	DIAGONAL 56	BARCELONA	936692866	ELISA DAPENA	NULL
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	927785235	LAURA CARRASCO	NULL
CT24	SÁEZ Y CÍA	INFANTA MERCEDES 23	SEVILLA	954869234	MANUEL GUERRA	NULL
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ	NULL
CT26	FERRETERÍA LA ESCOBA	ORENSE 7	MADRID	918459346	JOSÉ GARCÍA	NULL
CT27	JUGUETES EL BARATO	VÍA AUGUSTA 245	BARCELONA	933486984	ELVIRA IGLESIAS	NULL
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ	NULL
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	912495973	ADRIÁN ÁLVAREZ	NULL
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL	NULL
CT32	DEPORTES NAUTICOS GARCÍA	JUAN FERNÁNDEZ 89	ÁVILA	920268648	JUAN CONRADO	NULL
CT33	CONFECCIONES RUIZ	LLOBREGAT 345	BARCELONA	934587615	CARLOS SANZ	NULL
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	915483627	ANGEL SANTAMARÍA	NULL
CT35	FERRETERÍA EL MARTILLO	CASTELLANOS 205	SALAMANCA	923548965	JOAQUÍN FERNANDEZ	NULL
CT36	JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID	916872354	PEDRO IGLESIAS	NULL
CT37	ALMACENES FERNANDEZ	ANTÓN 67	TERUEL	978564025	MARIA ARDANZA	NULL
CT38	CONFECCIONES MÓNICA	MOTORES 67	BARCELONA	935681245	PEDRO SERRANO	NULL
CT39	FERRETERÍA LIMA	VALLECAS 45	MADRID	913532785	LUIS GARCÍA	NULL
CT40	DEPORTES EL BRASILEÑO	ABEL MARTÍNEZ 67	SALAMANCA	921548762	CARLOS GÓMEZ	NULL

Por otro lado tenemos al tabla PEDIDOS.

NÚMERO DE PEDIDO	CÓDIGO CLIENTE	FECHA DE PEDIDO	FORMA DE PAGO	DESCUENTO	ENVIADO
1	CT01	11/03/2000	CONTADO	0.02	VERDADERO
3	CT23	18/03/2000	APLAZADO	0.06	FALSO
5	CT25	31/03/2000	CONTADO	0.09	FALSO
7	CT12	12/04/2000	CONTADO	0.07	FALSO
8	CT01	15/04/2000	TARJETA	0.02	VERDADERO
9	CT21	21/04/2000	CONTADO	0.04	FALSO
13	CT13	30/04/2000	APLAZADO	0.03	FALSO
22	CT07	31/05/2000	TARJETA	0.05	VERDADERO
25	CT18	02/06/2000	CONTADO	0.06	FALSO
27	CT34	06/06/2000	CONTADO	0.04	FALSO
31	CT30	08/06/2000	TARJETA	0.05	VERDADERO
28	CT28	08/06/2000	APLAZADO	0.08	FALSO
47	CT34	31/07/2000	APLAZADO	0.08	FALSO
30	CT02	15/08/2000	CONTADO	0.06	VERDADERO
63	CT28	10/09/2000	CONTADO	0.09	FALSO
77	CT01	28/10/2000	CONTADO	0.05	FALSO
79	CT34	12/12/2000	CONTADO	0.05	FALSO
105	CT30	01/01/2001	APLAZADO	0.09	FALSO
102	CT06	12/01/2001	CONTADO	0.07	VERDADERO
103	CT02	24/01/2001	CONTADO	0.04	FALSO
29	CT30	02/04/2001	TARJETA	0.06	FALSO
11	CT04	01/05/2001	CONTADO	0.08	VERDADERO
16	CT25	11/05/2001	CONTADO	0.12	FALSO
12	CT06	19/05/2001	CONTADO	0.09	VERDADERO
21	CT16	28/05/2001	CONTADO	0.03	FALSO

26	CT09	04/06/2001	APLAZADO	0.07	VERDADERO
32	CT14	20/06/2001	APLAZADO	0.06	FALSO
35	CT26	30/06/2001	CONTADO	0.06	FALSO
37	CT24	02/07/2001	TARJETA	0.03	VERDADERO
39	CT20	08/07/2001	TARJETA	0.06	VERDADERO
43	CT09	18/07/2001	CONTADO	0.07	FALSO
73	CT01	02/08/2001	CONTADO	0.07	FALSO
86	CT09	24/12/2001	APLAZADO	0.03	FALSO
98	CT01	27/12/2001	CONTADO	0.08	VERDADERO
5050	CT30	27/03/2002	TARJETA	0.00	VERDADERO
19	CT10	22/05/2002	CONTADO	0.07	VERDADERO
34	CT26	23/06/2002	TARJETA	0.05	FALSO
40	CT04	12/07/2002	CONTADO	0.12	FALSO
42	CT34	15/07/2002	APLAZADO	0.07	VERDADERO
44	CT34	20/07/2002	APLAZADO	0.04	FALSO
45	CT30	22/07/2002	TARJETA	0.07	FALSO
46	CT31	25/07/2002	CONTADO	0.06	FALSO
5005	CT30	10/08/2002	TARJETA	0.00	VERDADERO
72	CT01	18/08/2002	CONTADO	0.05	VERDADERO
48	CT18	30/08/2002	CONTADO	0.03	FALSO
49	CT28	02/09/2002	CONTADO	0.03	FALSO
50	CT09	05/09/2002	APLAZADO	0.08	FALSO
51	CT09	05/09/2002	CONTADO	0.05	VERDADERO
74	CT01	17/09/2002	APLAZADO	0.08	FALSO
75	CT01	30/09/2002	TARJETA	0.12	FALSO
76	CT01	19/10/2002	CONTADO	0.04	VERDADERO
85	CT04	23/12/2002	TARJETA	0.04	FALSO

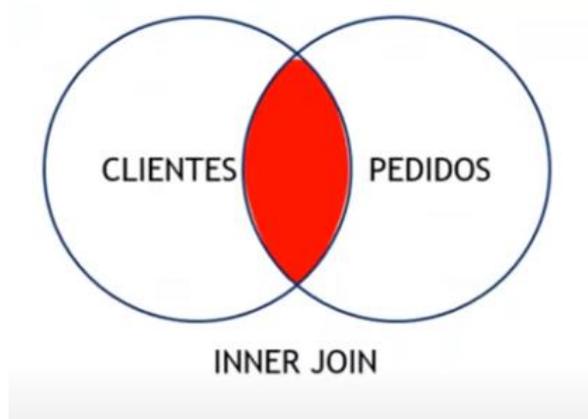
Lo que estamos viendo es que un mismo cliente me puede realizar varios pedidos.
La relación que existe en estas dos tablas es la siguiente.



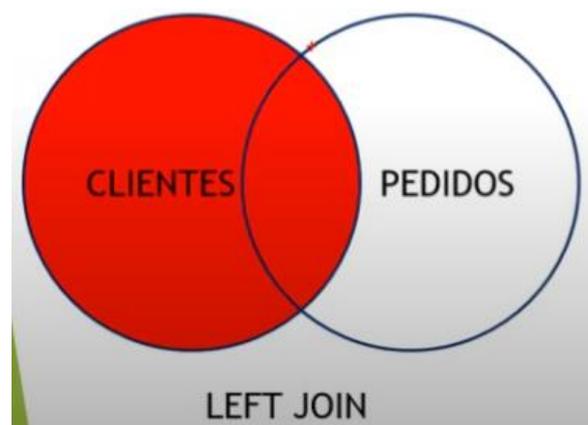
Hay una relación de uno a varios, un cliente por varios pedidos.
También pueden haber relaciones de uno a uno y otras de varios a varios, pero en este caso la relación es de uno a varios.
Utilizando los JOINS puedo hacer consultas sobre tablas relacionadas.

INNER JOIN: Nos va a dar la información que hay en varias tablas, los clientes que están en las dos tablas.

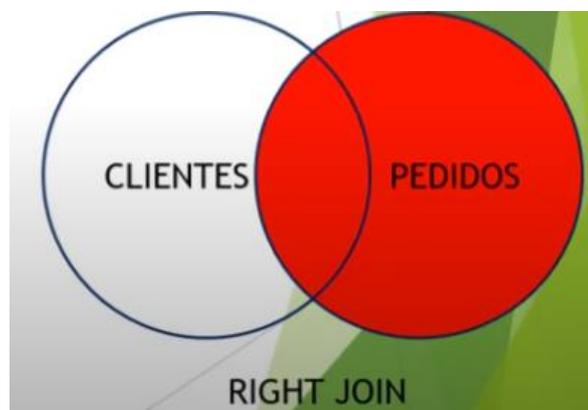
Los clientes han hecho pedidos.



LEFT JOIN: No solamente nos muestra los clientes que han hecho pedidos, además los clientes que no han hecho pedidos.



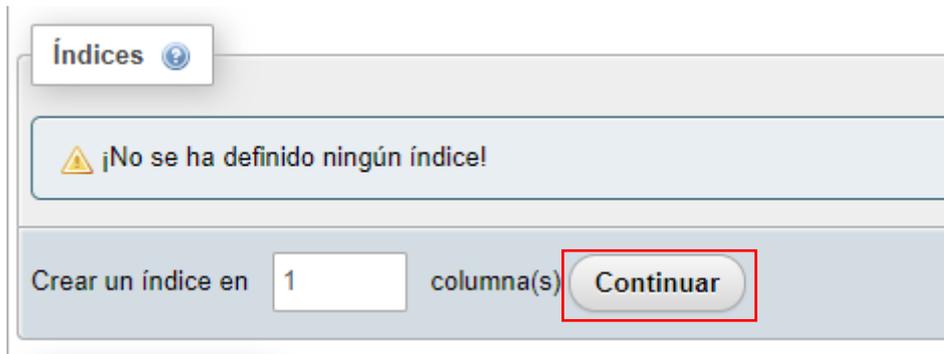
RIGHT JOIN: Nos devuelven todos los registros de clientes que ha hecho pedido además de aquellos pedidos que aún no están asignado a clientes.



Capítulo 9.- Consultas Multitabla III. Left Join y Right Join

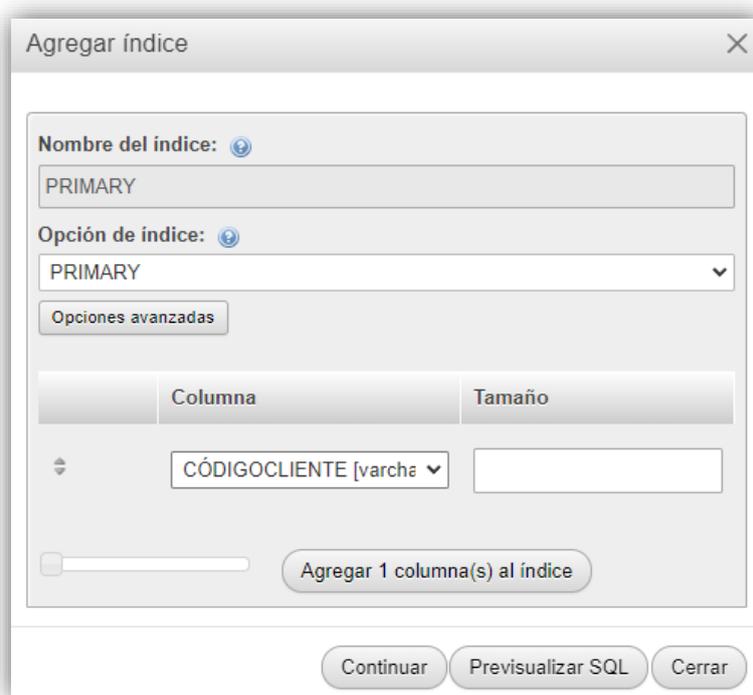
Una vez tenemos las tablas, las tenemos que relacionar.

Lo primero vamos a seleccionar la tabla clientes y seleccionamos la pestaña estructura.



Hay un apartado donde pone índices.

En este caso vamos a crear un índice de 1 columna seguido del botón continuar.



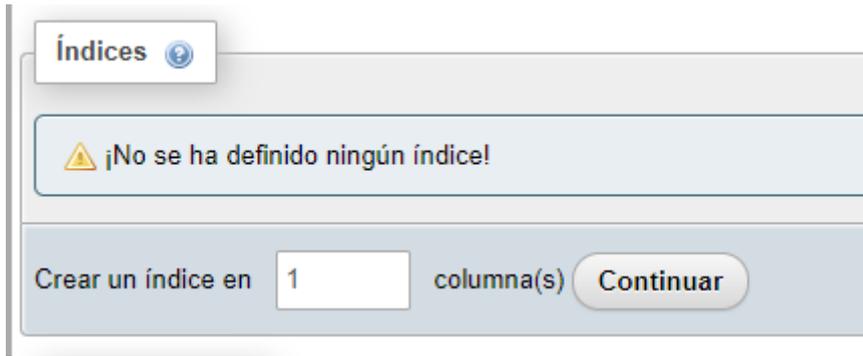
En opción de índice seleccionaremos PRIMARY y en columna seleccionaremos CÓDIGOCIENTE seguido del botón continuar.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Si	No	CÓDIGOCIENTE	38	A	No	

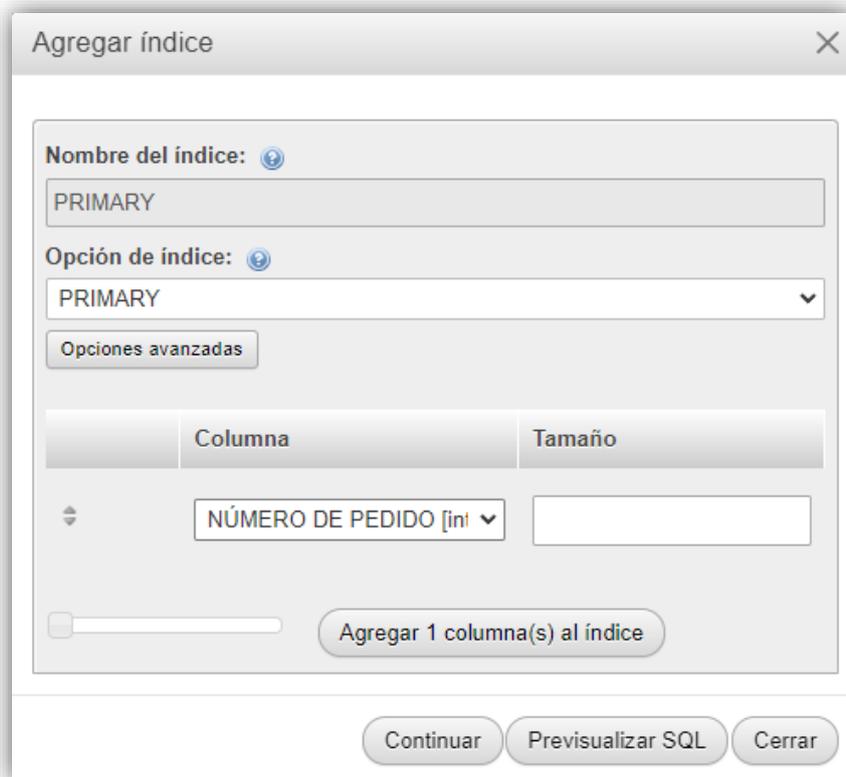
Ya hemos creado el índice de la tabla clientes.

Vamos a hacer lo mismo con la tabla de PEDIDOS.

Seleccionamos PEDIDOS y seleccionamos la pestaña Estructura.



Le damos a continuar.

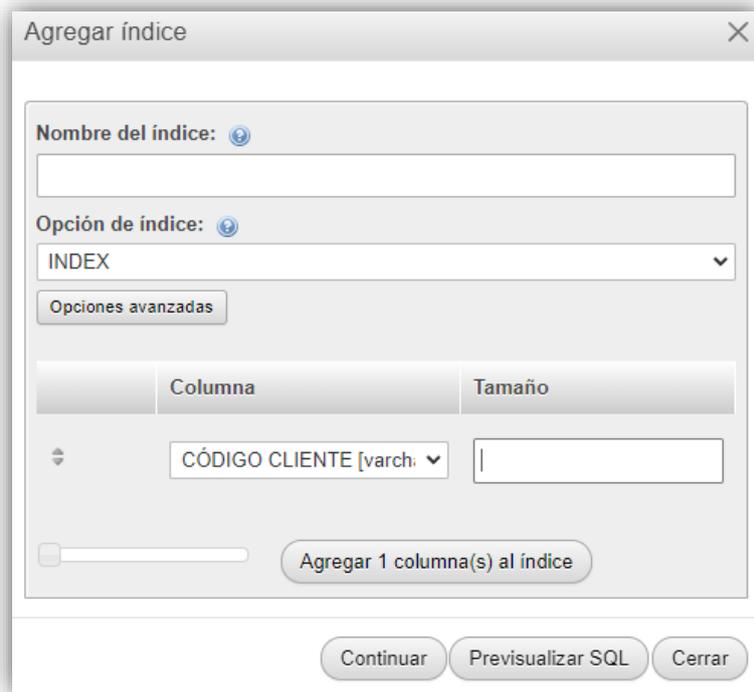


Como opción de índice seleccionaremos PRIMARY y en columna seleccionaremos NÚMERO DE PEDIDO, seguido del botón continuar.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	NÚMERO DE PEDIDO	52	A	No	

Ya hemos creado el índice.

Vamos a crear un segundo índice:

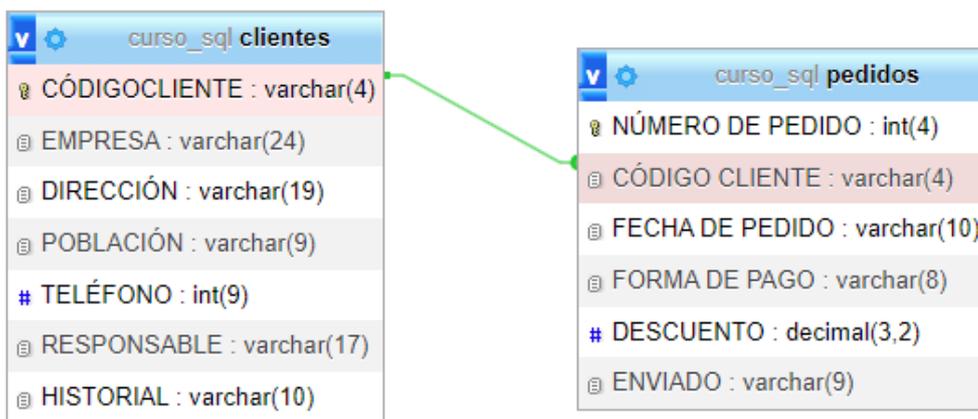


En opción de índice seleccionaremos INDEX y en columna CÓDIGO CLIENTE que será el campo que relacionaremos, seguido de continuar.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	NÚMERO DE PEDIDO	52	A	No	
Editar Renombrar Eliminar	CÓDIGO CLIENTE	BTREE	No	No	CÓDIGO CLIENTE	52	A	Sí	

Ya tenemos el segundo índice.

Seleccionamos el nombre de la base de datos y a continuación la pestaña Diseñador.



Hacemos la relación.

Vamos a hacer una consulta de todos los pedidos que han hecho los clientes de Madrid.

```

1 SELECT CLIENTES.CÓDIGOCLIENTE, POBLACIÓN, DIRECCIÓN, NÚMEROPEPIDI, PEDIDOS.CÓDIGOCLIENTE, FORMADEPAGO
2 FROM CLIENTES INNER JOIN PEDIDOS ON CLIENTES.CÓDIGOCLIENTE=PEDIDOS.CÓDIGOCLIENTE
3 WHERE POBLACIÓN= 'MADRID'

```

Este será el resultado:

CÓDIGOCLIENTE	POBLACIÓN	DIRECCIÓN	NÚMEROPEPIDI	CÓDIGOCLIENTE	FORMADEPAGO
CT01	MADRID	LAS FUENTES 78	1	CT01	CONTADO
CT23	MADRID	ORTEGA Y GASSET 129	3	CT23	APLAZADO
CT01	MADRID	LAS FUENTES 78	8	CT01	TARJETA
CT04	MADRID	VALLECAS 34	11	CT04	CONTADO
CT10	MADRID	PASEO DE ÁLAMOS 78	19	CT10	CONTADO
CT07	MADRID	AMÉRICA 45	22	CT07	TARJETA
CT34	MADRID	CASTILLA Y LEÓN 34	27	CT34	CONTADO
CT30	MADRID	ATOCHA 55	29	CT30	TARJETA
CT30	MADRID	ATOCHA 55	31	CT30	TARJETA
CT14	MADRID	GUZMÁN EL BUENO 45	32	CT14	APLAZADO
CT26	MADRID	ORENSE 7	34	CT26	TARJETA
CT26	MADRID	ORENSE 7	35	CT26	CONTADO
CT04	MADRID	VALLECAS 34	40	CT04	CONTADO
CT34	MADRID	CASTILLA Y LEÓN 34	42	CT34	APLAZADO
CT34	MADRID	CASTILLA Y LEÓN 34	44	CT34	APLAZADO
CT30	MADRID	ATOCHA 55	45	CT30	TARJETA
CT34	MADRID	CASTILLA Y LEÓN 34	47	CT34	APLAZADO
CT01	MADRID	LAS FUENTES 78	72	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	73	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	74	CT01	APLAZADO
CT01	MADRID	LAS FUENTES 78	75	CT01	TARJETA
CT01	MADRID	LAS FUENTES 78	76	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	77	CT01	CONTADO
CT34	MADRID	CASTILLA Y LEÓN 34	79	CT34	CONTADO
CT04	MADRID	VALLECAS 34	85	CT04	TARJETA

Vamos a cambiar por un LEFT JOIN.

```

1 SELECT CLIENTES.CÓDIGOCLIENTE, POBLACIÓN, DIRECCIÓN, NÚMEROPEPIDI, PEDIDOS.CÓDIGOCLIENTE, FORMADEPAGO
2 FROM CLIENTES LEFT JOIN PEDIDOS ON CLIENTES.CÓDIGOCLIENTE=PEDIDOS.CÓDIGOCLIENTE
3 WHERE POBLACIÓN= 'MADRID'

```

Este será el resultado:

CÓDIGOCLIENTE	POBLACIÓN	DIRECCIÓN	NÚMEROPEPIDI	CÓDIGOCLIENTE	FORMADEPAGO
CT01	MADRID	LAS FUENTES 78	1	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	8	CT01	TARJETA
CT01	MADRID	LAS FUENTES 78	72	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	73	CT01	CONTADO
CT01	MADRID	LAS FUENTES 78	74	CT01	APLAZADO
CT01	MADRID	LAS FUENTES 78	75	CT01	TARJETA
CT01	MADRID	LAS FUENTES 78	76	CT01	CONTADO

CT01	MADRID	LAS FUENTES 78	98	CT01	CONTADO
CT04	MADRID	VALLECAS 34	11	CT04	CONTADO
CT04	MADRID	VALLECAS 34	40	CT04	CONTADO
CT04	MADRID	VALLECAS 34	85	CT04	TARJETA
CT07	MADRID	AMÉRICA 45	22	CT07	TARJETA
CT10	MADRID	PASEO DE ÁLAMOS 78	19	CT10	CONTADO
CT14	MADRID	GUZMÁN EL BUENO 45	32	CT14	APLAZADO
CT19	MADRID	FUENCARRAL 78	NULL	NULL	NULL
CT23	MADRID	ORTEGA Y GASSET 129	3	CT23	APLAZADO
CT26	MADRID	ORENSE 7	34	CT26	TARJETA
CT26	MADRID	ORENSE 7	35	CT26	CONTADO
CT30	MADRID	ATOCHA 55	29	CT30	TARJETA
CT30	MADRID	ATOCHA 55	31	CT30	TARJETA
CT30	MADRID	ATOCHA 55	45	CT30	TARJETA
CT30	MADRID	ATOCHA 55	105	CT30	APLAZADO
CT30	MADRID	ATOCHA 55	5005	CT30	TARJETA
CT30	MADRID	ATOCHA 55	5050	CT30	TARJETA
CT34	MADRID	CASTILLA Y LEÓN 34	27	CT34	CONTADO
CT34	MADRID	CASTILLA Y LEÓN 34	42	CT34	APLAZADO
CT34	MADRID	CASTILLA Y LEÓN 34	44	CT34	APLAZADO
CT34	MADRID	CASTILLA Y LEÓN 34	47	CT34	APLAZADO
CT34	MADRID	CASTILLA Y LEÓN 34	79	CT34	CONTADO
CT36	MADRID	ORENSE 89	NULL	NULL	NULL
CT39	MADRID	VALLECAS 45	NULL	NULL	NULL

Además de los clientes de Madrid que han hecho pedido, también salen los clientes de Madrid que no han hecho pedido.

Ahora queremos consultar por los clientes de Madrid que no han hecho ningún pedido.

```

1 SELECT CLIENTES.CÓDIGOCLIENTE, POBLACIÓN, DIRECCIÓN, NÚMERODEPEDIDO, PEDIDOS.CÓDIGOCLIENTE, FORMADEPAGO
2 FROM CLIENTES LEFT JOIN PEDIDOS ON CLIENTES.CÓDIGOCLIENTE=PEDIDOS.CÓDIGOCLIENTE
3 WHERE POBLACIÓN= 'MADRID' AND PEDIDOS.CÓDIGOCLIENTE IS NULL

```

Este será el resultado:

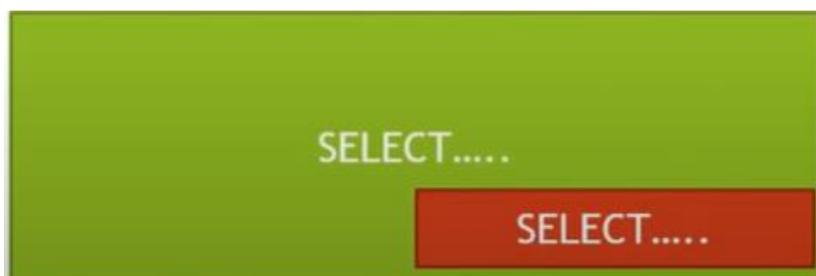
CÓDIGOCLIENTE	POBLACIÓN	DIRECCIÓN	NÚMERODEPEDIDO	CÓDIGOCLIENTE	FORMADEPAGO
CT19	MADRID	FUENCARRAL 78	NULL	NULL	NULL
CT36	MADRID	ORENSE 89	NULL	NULL	NULL
CT39	MADRID	VALLECAS 45	NULL	NULL	NULL

Capítulo 10.- Subconsultas I

En este capítulo veremos...

- Subconsultas ¿Qué son? Tipos
 - Subconsulta escalonada
 - Subconsulta de lista
 - Subconsulta correlacionada

Una subconsulta no es ni más ni menos una consulta dentro de otra.



Un SELECT dentro de otro SELECT.

Lo podemos considerar como un SELECT padre y un SELECT hijo.

El SELECT hijo tiene que sintácticamente entre paréntesis y se va a utilizar como filtro o criterio de lo que sería el SELECT padre.

En definitiva lo que estamos haciendo con una subconsulta es comparar los registros que devuelve el SELECT interior o SELECT hijo con lo que pretendemos obtener del SELECT principal o padre.

Operadores de comparación:

- Lógicos
 - AND: todas las condiciones verdaderas
 - OR: una de las condiciones verdaderas
 - NOT: valor contrario de la expresión
- Comparación
 - LIKE: comparación de cadenas de caracteres
 - <>, <=, >=, <, >: mayor que..., menor que..., distinto que... etc.
 - BETWEEN: intervalos
 - **IN**
 - **ANY**
 - **ALL**

Las subconsultas escalonadas lo que tienen es que la consulta interna devuelve una única columna con un único registro y esto se utiliza como criterio o como comparación en la consulta padre.

Veamos un ejemplo:

Queremos realizar una consulta que nos devuelva el nombre del producto y la sección de aquellos productos cuyo precio sea superior a la media.

Por un lado habría que realizar dos consultas una primera que nos muestre la media del precio de todos los productos que tenemos en la tabla PRODUCTOS y una vez que obtenemos la media utilizarlo como criterio para obtener el nombre y sección de los artículos cuyo precio sea superior con el dato de la consultar interna o hijo.

Primero vamos a realizar una consulta para obtener la media de los artículos de esta tabla.

```
1 SELECT AVG(PRECIO) FROM PRODUCTOS
```

Este será el resultado:

```
AVG(PRECIO)
12648.60777250
```

Ya tenemos la subconsulta escalonada.

Este dato lo vamos a utilizar para filtrar o comparar con la consulta padre.

```
1 SELECT NOMBREARTÍCULO, SECCIÓN FROM PRODUCTOS WHERE PRECIO >
2 (SELECT AVG(PRECIO) FROM PRODUCTOS)
```

← Padre
← Hijo

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN
ABRIGO CABALLERO	CONFECCIÓN

```
1 SELECT NOMBREARTÍCULO, SECCIÓN FROM PRODUCTOS WHERE PRECIO <
2 (SELECT AVG(PRECIO) FROM PRODUCTOS)
```

↖

Ahora queremos consultar por todos los artículos que son menores a la media.

Este será el resultado:

NOMBREARTÍCULO	SECCIÓN
DESTORNILLADOR	FERRETERÍA
TRAJE CABALLERO	CONFECCIÓN
COCHE TELEDIRIGIDO	JUGUETERÍA
RAQUETA TENIS	DEPORTES
MANCUERNAS	DEPORTES
SERRUCHO	CONFECCIÓN
CORREPASILLOS	JUGUETERÍA
PANTALÓN SEÑORA	CONFECCIÓN
CONSOLA VIDEO	JUGUETERÍA
TUBOS	CERÁMICA

Los mostrará todos menos el abrigo de caballero.

(...)

La subconsulta de lista la diferencia entre las dos es que en lugar de devolver un único registro nos devuelve una lista de registros.

Con este tipo de subconsulta se suelen utilizar los operadores mencionados anteriormente IN, ANY y ALL.

Imaginemos que queremos hacer una consulta que nos devuelva los artículos cuyo precio superior de todos los artículos de CERÁMICA.

Primero tenemos que hacer una subconsulta que nos diga el precio de todos los artículos de CERÁMICA, para obtener el precio superior.

```
1 SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = 'CERÁMICA'
```

Esto nos retorna:

PRECIO
168.4253
54.0911
43.2728
19.7468
29.0434
127.7704

```
1 SELECT *FROM PRODUCTOS WHERE PRECIO > ALL  
2 (SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = 'CERÁMICA')
```

Utilizamos el operador ALL este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	VERDADERO	ITALIA	NULL
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	VERDADERO	MARRUECOS	NULL
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA	NULL
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	VERDADERO	JAPÓN	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360.0736	2001-05-03	VERDADERO	MARRUECOS	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL

Si cambiamos por ANY significa mayor a cualquier valor de la lista.

```
1 SELECT *FROM PRODUCTOS WHERE PRECIO > ANY  
2 (SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = 'CERÁMICA')
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	VERDADERO	ITALIA	NULL
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	VERDADERO	MARRUECOS	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR07	CONFECCIÓN	SERRUCHO	30.2045	2001-03-23	VERDADERO	FRANCIA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPÓN	NULL
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	VERDADERO	MARRUECOS	NULL
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23	VERDADERO	USA	NULL
AR13	CONFECCIÓN	CAMISA CABALLERO	67.1306	2002-08-11	FALSO	ESPAÑA	NULL
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	VERDADERO	JAPÓN	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA	NULL
AR17	JUGUETERÍA	MUÑECA ANDADORA	105.0593	2001-01-04	FALSO	ESPAÑA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA	NULL
AR19	CONFECCIÓN	BLUSA SRA.	101.0566	2000-03-18	VERDADERO	CHINA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	VERDADERO	CHINA	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR26	JUGUETERÍA	FUERTE DE SOLDADOS	143.7020	2000-11-25	VERDADERO	JAPÓN	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360.0736	2001-05-03	VERDADERO	MARRUECOS	NULL
AR31	JUGUETERÍA	PISTOLA CON SONIDOS	57.2500	2001-04-15	FALSO	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	FALSO	ESPAÑA	NULL
AR34	OFICINA	PIE DE LÁMPARA	39.7606	2001-05-27	VERDADERO	TURQUÍA	NULL
AR35	FERRETERÍA	LIMA GRANDE	22.0692	2002-08-10	FALSO	ESPAÑA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA	NULL

La diferencia es notable.

Otro ejemplo, vamos a hacer una subconsulta que nos muestre en principio todos los artículos de la tabla productos cuyo precio sea superior a todos los artículos de JUGUETERÍA y luego cambiaremos todos por cualquier artículo de JUGUETERÍA.

```
1 SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = "JUGUETERÍA"
```

Este será el resultado:

PRECIO
159.4462
103.3356

442.5444

1505.3766

105.0593

143.7020

57.2500

Fijaros que hay un artículo que cuesta 1.505,3766 €.

```
1 SELECT * FROM PRODUCTOS WHERE PRECIO > ALL
2 (SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = "JUGUETERÍA")
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL

Si cambiamos ALL por ANY

```
1 SELECT * FROM PRODUCTOS WHERE PRECIO > ANY
2 (SELECT PRECIO FROM PRODUCTOS WHERE SECCIÓN = "JUGUETERÍA")
```

Este será el resultado:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	VERDADERO	ITALIA	NULL
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	VERDADERO	MARRUECOS	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPÓN	NULL
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	VERDADERO	MARRUECOS	NULL
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA	NULL
AR13	CONFECCIÓN	CAMISA CABALLERO	67.1306	2002-08-11	FALSO	ESPAÑA	NULL
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	VERDADERO	JAPÓN	NULL
AR17	JUGUETERÍA	MUÑECA ANDADORA	105.0593	2001-01-04	FALSO	ESPAÑA	NULL
AR19	CONFECCIÓN	BLUSA SRA.	101.0566	2000-03-18	VERDADERO	CHINA	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522.6930	2001-07-10	VERDADERO	ITALIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN	NULL
AR26	JUGUETERÍA	FUERTE DE SOLDADOS	143.7020	2000-11-25	VERDADERO	JAPÓN	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	VERDADERO	ITALIA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360.0736	2001-05-03	VERDADERO	MARRUECOS	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	VERDADERO	CHINA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA	NULL

Al poner ANY se mostrará todos los artículos superiores al artículo con menor precio de la sección de JUGUETERÍA.

Capítulo 11.- Subconsultas II

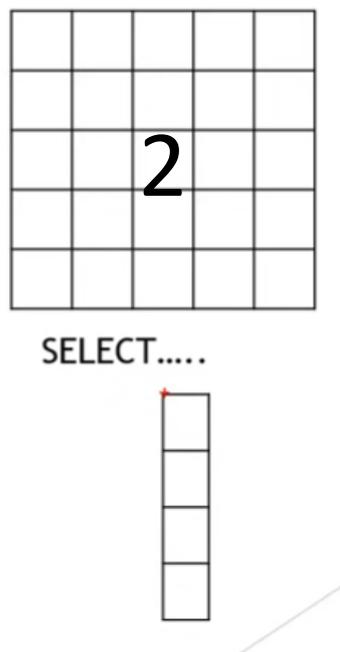
Vamos a seguir con las subconsultas.

En este capítulo veremos...

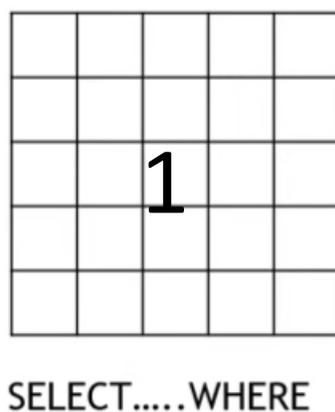
- Predicados IN y NOT IN

Los podemos traducir como están y no están.

Ya sabemos que las subconsultas lo que hacen es filtrar la consulta principal en base a los registros obtenidos en una subconsulta.

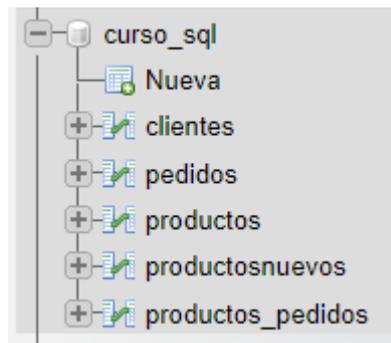


Tenemos dos tablas la tabla1 principal tabla2 la secundaria, realizamos una consulta sobre la tabla2 y en esta ocasión esta consulta nos va a devolver una única columna y esa columna de resultados va a ser la que se utilice como criterio para la consulta principal.



Nos retornará una sola columna servirá como criterio para la consulta principal.

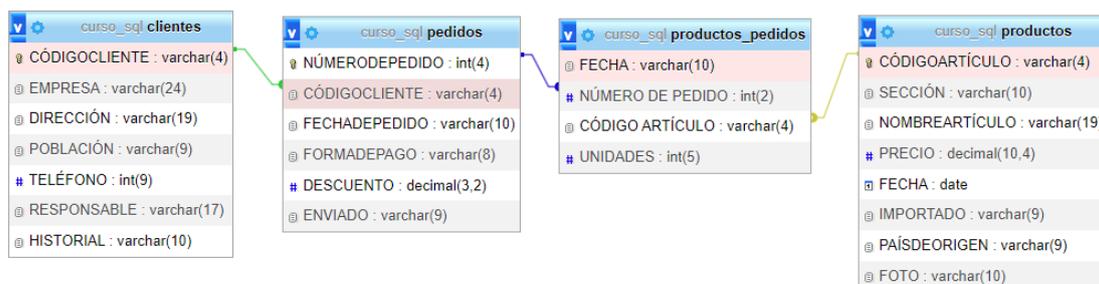
Y para ello utilizaremos IN / NOT IN, están o no están.



Tenemos unos CLIENTES que hacen una serie de pedidos, esos PEDIDOS contienen una serie de PRODUCTOS que son los que se piden y luego tenemos una tabla llamada PRODUCTOS PEDIDOS que es una unión entre la tabla de PROCTOS y la tabla PEDIDOS.

FECHA	NÚMERO DE PEDIDO	CÓDIGO ARTÍCULO	UNIDADES
2023-03-02	1	AR01	11
2023-03-02	1	AR04	10
2023-03-02	1	AR15	4
2023-03-02	1	AR22	18
2023-03-02	3	AR02	20
2023-03-02	3	AR22	3

La tabla PRODUCTOSPEDIDOS, nos indica la fecha, el número de pedido , código del clientes y las unidades pedidas.



Vamos a crear una subconsulta utilizando el predicado IN.

Podemos realizar una consulta que nos devuelva el nombre y precio de aquellos productos de los que se ha pedido más de 20 unidades.

Necesito información de dos tablas de PRODUCTOS para averiguar el nombre y el precio, pero también necesito información de la tabla PRODUCTOSPEDIDOS para ver de esos artículos cuales son aquellos que tienen más de 20 unidades.

```

1 SELECT NOMBREARTICULO, PRECIO FROM PRODUCTOS WHERE CODIGOARTICULO IN
2 (SELECT CODIGOARTICULO FROM PRODUCTOS_PEDIDOS WHERE UNIDADES > 20)

```

Este será el resultado:

NOMBREARTICULO	PRECIO
LLAVE INGLESA	24.3986
CENICERO	19.7468
MARTILLO	11.3952
CAZADORA PIEL	522.6930
BALÓN RUGBY	111.6440
BALÓN FÚTBOL	43.9147
ABRIGO SRA	360.0736
PISTOLA CON SONIDOS	57.2500
LIMA GRANDE	22.0692
JUEGO DE BROCAS	15.0950

Vamos a ver otra forma de realizar la consulta:

```
1 SELECT NOMBREARTICULO, PRECIO FROM PRODUCTOS INNER JOIN
2 PRODUCTOS_PEDIDOS ON
3 PRODUCTOS.CODIGOARTICULO=PRODUCTOS_PEDIDOS.CODIGOARTICULO
4 WHERE UNIDADES > 20
```

Este será el resultado:

NOMBREARTICULO	PRECIO
LLAVE INGLESA	24.3986
LIMA GRANDE	22.0692
BALÓN FÚTBOL	43.9147
CENICERO	19.7468
CAZADORA PIEL	522.6930
BALÓN RUGBY	111.6440
ABRIGO SRA	360.0736
PISTOLA CON SONIDOS	57.2500
MARTILLO	11.3952
JUEGO DE BROCAS	15.0950

También tenemos el predicado NOT IN (no está en).

Por ejemplo vamos a elaborar una subconsulta que me muestre aquellos clientes que no han pagado con tarjeta o que no han realizado pedidos.

```
1 SELECT EMPRESA, POBLACION FROM CLIENTES
2 WHERE CODIGOCLIENTE NOT IN
3 (SELECT CODIGOCLIENTE FROM PEDIDOS
4 WHERE FORMADEPAGO="TARJETA")
```

Vamos a analizar la subconsulta:

Seleccionamos el campo CODIGOCLIENTE de la tabla PEDIDOS, DONDE FORMADEPAGO se igual a "TARJETA".

Esta consulta nos retorna una lista de los clientes que han pagado con tarjeta.

```
1 SELECT EMPRESA, POBLACION FROM CLIENTES
2 WHERE CODIGOCLIENTE NOT IN
3 (SELECT CODIGOCLIENTE FROM PEDIDOS
4 WHERE FORMADEPAGO="TARJETA")
```

Vamos a analizar la consulta principal:

Seleccionamos el campo EMPRESA, POBLACION de la tabla CLIENTES donde CODIGOCLIENTE no está en la lista que ha generado la subconsulta.

Capítulo 12.- Consultas de acción I

En este capítulo veremos ...

- Consultas de acción:
 - Actualización
 - Creación de tablas
 - Eliminación
 - Datos anexados
- Comandos DML y DDL
 - Create
 - Update
 - Delete
 - Insert Into
 - Select into

Son aquellas que modifican la información de las tablas.

Se pueden modificar de varias formas, o bien actualizando la información, creando tablas nuevas a partir de otras existentes, eliminando información dentro de las tablas o bien anexando información, registros que tenemos almacenados en una tabla unirlos a los registros que tenemos en otra tabla.

Supongamos que en la tabla PRODUCTOS queremos incrementar el precio a los artículos pertenecientes a una sección, por ejemplo a la SECCIÓN DE DEPORTES. Además es un trabajo que se suele realizar cada 2 o 3 semanas.

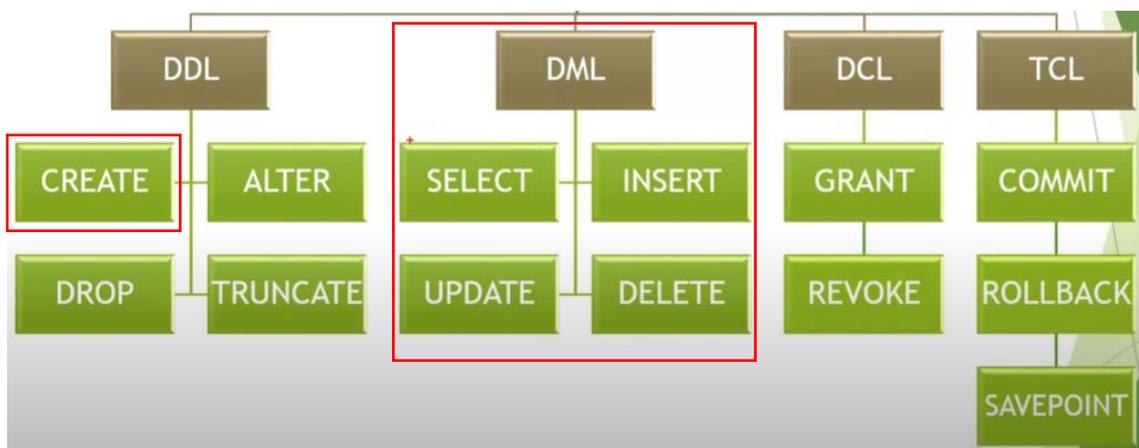
Las consultas de creación de tablas lo que hace es crear una tabla nueva a partir de la información que tengamos en otra tabla, por ejemplo imaginaros de la tabla de CLIENTES la delegación de Barcelona nos pide los CLIENTES que son de Barcelona.

La consulta de eliminación va a eliminar una serie de registros en base a un criterio o no, que tengamos dentro de una tabla, imaginaros que ya no vamos a trabajar con los clientes de Barcelona y los queremos eliminar.

Las consultas de datos anexados imaginaros el caso de que alguien nos da una tabla con información que no tenemos en nuestra base de datos y queremos unir la información de esta tabla nueva en otra tabla que ya tengamos nosotros.

Para realizar este tipo de consultas vamos a tener que utilizar comandos DML y DDL con las siguientes instrucciones: Create, Update, Delete, Insert Into y Select into.

La consulta de creación de tablas no se hace igual entre los gestores de base de datos.



Imaginaros que nos piden incrementar de la tabla PRODUCTOS en 10 euros todos los PRECIOS de la SECCION de DEPORTES.

```
1 UPDATE PRODUCTOS SET PRECIO=PRECIO+10 WHERE SECCION = 'DEPORTES'
```

Le damos al botón continuar.

✓ 10 filas afectadas. (La consulta tardó 0,0003 segundos.)

Ya tenemos los presiones actualizados.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN
AR04	DEPORTES	RAQUETA TENIS	103.4694	2000-03-20	VERDADERO	USA
AR06	DEPORTES	MANCUERNAS	70.0000	2000-09-13	VERDADERO	USA
AR18	DEPORTES	PISTOLA OLÍMPICA	56.7347	2001-02-02	VERDADERO	SUECIA
AR24	DEPORTES	BALÓN RUGBY	121.6440	2000-11-11	VERDADERO	USA
AR25	DEPORTES	BALÓN BALONCESTO	85.2731	2001-06-25	VERDADERO	JAPÓN
AR28	DEPORTES	BALÓN FÚTBOL	53.9147	2002-07-04	FALSO	ESPAÑA
AR32	DEPORTES	CRONÓMETRO	449.1764	2002-01-03	VERDADERO	USA
AR38	DEPORTES	CAÑA DE PESCA	280.0000	2000-02-14	VERDADERO	USA
AR40	DEPORTES	BOTA ALPINISMO	154.0000	2002-05-05	FALSO	ESPAÑA
AR41	DEPORTES	PALAS DE PING PONG	31.6000	2002-02-02	FALSO	ESPAÑA

En el caso de que te arrepientas con los cambios no hay vuelta atrás.

Si yo quiero volver atrás tendré que hacer otra consulta de actualización, pero en este caso restándole 10.

```
1 UPDATE PRODUCTOS SET PRECIO=PRECIO-10 WHERE SECCION='DEPORTES'
```

Le damos a continuar.

✓ 10 filas afectadas. (La consulta tardó 0,0003 segundos.)

Este será el resultado:

```
1 SELECT * FROM PRODUCTOS WHERE SECCION='DEPORTES'
```

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	VERDADERO	USA
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	VERDADERO	JAPÓN
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	FALSO	ESPAÑA
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	VERDADERO	USA
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	VERDADERO	USA
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	FALSO	ESPAÑA
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	FALSO	ESPAÑA

Ahora necesitamos cambiar la denominación de la sección en lugar de llamarse DEPORTE tiene que llamarse DEPORTIVOS.

```
1 UPDATE PRODUCTOS SET SECCION = 'DEPORTIVOS'  
2 WHERE SECCION = 'DEPORTES'
```

Le damos al botón continuar.

✓ 10 filas afectadas. (La consulta tardó 0,0003 segundos.)

Si consultamos la tabla de PRODUCTOS

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	FALSO	ESPAÑA
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	VERDADERO	ITALIA
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	VERDADERO	MARRUECOS
AR04	DEPORTIVOS	RAQUETA TENIS	93.4694	2000-03-20	VERDADERO	USA
AR06	DEPORTIVOS	MANCUERNAS	60.0000	2000-09-13	VERDADERO	USA
AR07	CONFECCIÓN	SERRUCHO	30.2045	2001-03-23	VERDADERO	FRANCIA
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	VERDADERO	JAPÓN
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	VERDADERO	MARRUECOS
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24	VERDADERO	USA
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	VERDADERO	CHINA
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23	VERDADERO	USA
AR13	CONFECCIÓN	CAMISA CABALLERO	67.1306	2002-08-11	FALSO	ESPAÑA
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	VERDADERO	JAPÓN
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	VERDADERO	CHINA
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA
AR17	JUGUETERÍA	MUÑECAANDADORA	105.0593	2001-01-04	FALSO	ESPAÑA
AR18	DEPORTIVOS	PISTOLA OLÍMPICA	46.7347	2001-02-02	VERDADERO	SUECIA

Observamos que a reemplazado DEPORTES por DEPORTIVOS.

Para el campo de IMPORTADO como es de tipo booleano como criterio utilizaríamos TRUE o FALSE, también para TRUE se puede utilizar -1 y para False 0.

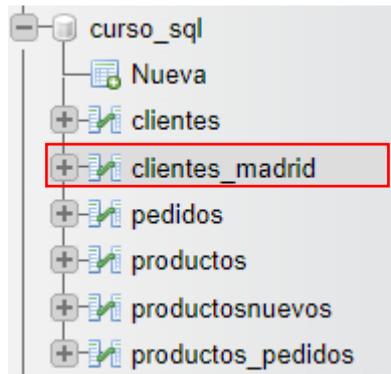
Vamos ahora con las consultas de creación de tabla.

Consiste en crear una tabla a partir de otra.

Imaginaros un compañero de trabajo necesita tener en la base de datos una tabla nueva que en estos momentos no existe con los CLIENTES de Madrid. Necesita trabajar en exclusiva con dichos clientes.

```
1 CREATE TABLE CLIENTES_MADRID SELECT * FROM  
2 CLIENTES WHERE POBLACION='MADRID'
```

Le damos al botón Continuar.



Ya se ha creado la tabla CLIENTES_MADRID y este será su contenido:

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTÍNEZ
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GÓMEZ
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	912649987	ELÍAS PÉREZ
CT10	FERRETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	914354866	MANUEL MENÉNDEZ
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	913299475	ANA JIMÉNEZ
CT19	CONFECCIONES GALÁN	FUENCARRAL 78	MADRID	913859234	JUAN GARCÍA
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	927785235	LAURA CARRASCO
CT26	FERRETERÍA LA ESCOBA	ORENSE 7	MADRID	918459346	JOSÉ GARCÍA
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	912495973	ADRIÁN ÁLVAREZ
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	915483627	ANGEL SANTAMARÍA
CT36	JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID	916872354	PEDRO IGLESIAS
CT39	FERRETERÍA LIMA	VALLECAS 45	MADRID	913532785	LUIS GARCÍA

La instrucción CREATE TABLE nos permite definir que claves va a tener la nueva tabla, propiedades de algunos campos, etc.

CONSULTAS DE ACCIÓN

CONSULTAS DE CREACIÓN DE TABLA:

1. Realizar una consulta de acción de creación de tabla a partir de la tabla **CLIENTES**, utilizando todos los campos de la tabla, pero únicamente los registros que sean de la población Madrid. El nuevo objeto lo nombramos con el texto "CLIENTES_DE_MADRID". Ejecutamos la consulta.
2. Realizar una consulta de acción de creación de tabla a partir de la tabla **PRODUCTOS**, utilizando todos los campos de la tabla, pero sólo los registros que sean de la sección DEPORTES. El nuevo objeto – tabla lo nombramos con el texto "ARTÍCULOS_DE_DEPORTES". Ejecutamos la consulta.
3. Realizar una consulta de acción de creación de tabla a partir de la tabla **PEDIDOS**, utilizando todos los campos de la tabla, pero sólo los registros que tengan registrada la forma de pago TARJETA. El nuevo objeto – tabla lo nombramos con el texto "PEDIDOS_PAGADOS_CON_TARJETA". Ejecutamos la consulta.

CONSULTA DE ACTUALIZACIÓN:

4. Realizar una consulta que actualice los precios de la tabla **ARTÍCULOS DE DEPORTE**. La actualización consiste en calcular el IVA (21%) y mostrar en ese campo como resultado el precio con el IVA incluido. Ejecutar la consulta.
5. Realizar una consulta que actualice el campo **DESCUENTO** de la tabla **PEDIDOS_PAGADOS_CON_TARJETA**. La actualización consiste poner a un 5% los descuentos que se muestran inferiores a esta cifra. Ejecutar la consulta.

- 1 CREATE TABLE CLIENTES_DE_MADRID SELECT * FROM CLINTES WHERE POBLACIÓN='MADRID'
- 2 CREATE TABLE ARTIFCULOS_DE_DEPORTES SELECT * FROM PRODUCTOS WHERE SECCIÓN='DEPORTES'
- 3 CREATE TABLE PEDIDOS_PAGADOS_CON_TARJETA SELECT * FROM PEDIDOS WHERE FORMADEPAGO='TARJETA'
- 4 UPDATE PRODUCTOS SET PRECIO=PRECIO*1.21 WHERE SECCIÓN='DEPORTES'
- 5 UPDATE PEDIDOS_PAGADOS_CON_TARJETA SET DESCUENTO=0.05 WHERE DESCUENTO<0.05

Capítulo 13.- Consultas de acción II Eliminación

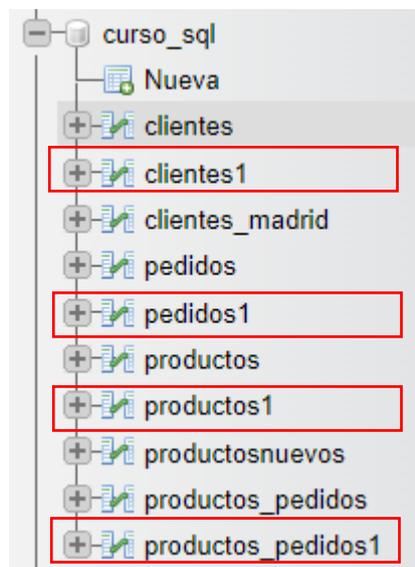
Para este capítulo vamos a duplicar las tablas y en las tablas duplicadas vamos a eliminar registros.

```
1 CREATE TABLE CLIENTES1 SELECT * FROM CLIENTES
```

```
1 CREATE TABLE PEDIDOS1 SELECT * FROM PEDIDOS
```

```
1 CREATE TABLE PRODUCTOS1 SELECT * FROM PRODUCTOS
```

```
1 CREATE TABLE PRODUCTOS_PEDIDOS1 SELECT * FROM PRODUCTOS_PEDIDOS
```



Las prácticas de este capítulo lo haremos con los archivos con terminación en 1.

Queremos eliminar de la tabla CLIENTES1 aquellos que son de Madrid.

```
1 DELETE FROM CLIENTES1 WHERE POBLACION='MADRID'
```

Le damos a continuar.

localhost dice

¿Realmente desea ejecutar "DELETE FROM CLIENTES1 WHERE POBLACION='MADRID' "?

Aceptar

Cancelar

Nos pide confirmación, le diremos Aceptar.

✓ 12 filas afectadas. (La consulta tardó 0,0018 segundos.)

Si abrimos la tabla CLIENTES1.

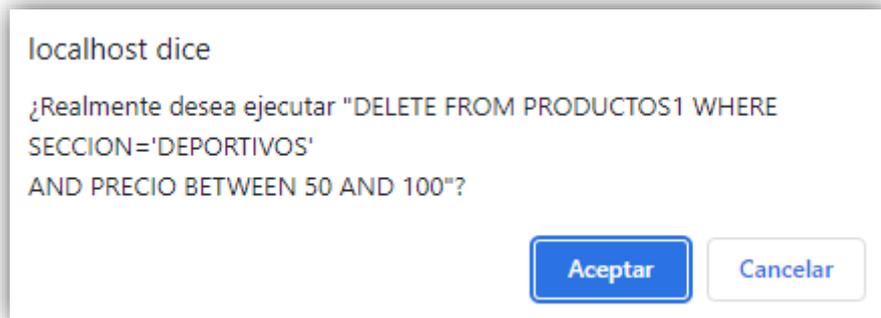
CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA
CT03	EL ESPAÑOLITO	MOTORES 34	BARCELONA	934565343	ANA FERNÁNDEZ
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ
CT08	JUGUETERÍA SUÁREZ	PARIS 123	BARCELONA	933457866	JUAN GARCÍA
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ
CT11	JUGUETES MARTÍNEZ	VIA LAYETANA 245	BARCELONA	936628554	FRANCISCO CUEVAS
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT15	EXCLUSIVAS FERNÁNDEZ	LLOBREGAT 250	BARCELONA	939558365	LUISA FERNÁNDEZ
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT17	BAZAR FRANCISCO	CARMEN 45	ZAMORA	980495288	CARLOS BELTRÁN
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT22	BAZAR EL BARAT	DIAGONAL 56	BARCELONA	936692866	ELISA DAPENA
CT24	SÁEZ Y CÍA	INFANTA MERCEDS 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ
CT27	JUGUETES EL BARATO	VÍA AUGUSTA 245	BARCELONA	933486984	ELVIRA IGLESIAS
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL
CT32	DEPORTES NAUTICOS GARCÍA	JUAN FERNÁNDEZ 89	ÁVILA	920268648	JUAN CONRADO
CT33	CONFECCIONES RUIZ	LLOBREGAT 345	BARCELONA	934587615	CARLOS SANZ
CT35	FERRETERÍA EL MARTILLO	CASTELLANOS 205	SALAMANCA	923548965	JOAQUÍN FERNANDEZ
CT37	ALMACENES FERNANDEZ	ANTÓN 67	TERUEL	978564025	MARIA ARDANZA
CT38	CONFECCIONES MÓNICA	MOTORES 67	BARCELONA	935681245	PEDRO SERRANO
CT40	DEPORTES EL BRASILEÑO	ABEL MARTÍNEZ 67	SALAMANCA	921548762	CARLOS GÓMEZ

Observamos que ya no tenemos ningún cliente de Madrid.

Imaginemos que queremos eliminar los artículos de la tabla PRODUCTOS1 que sean de la sección de DEPORTIVOS y su precio está entre 50 y 100 euros.

```
1 DELETE FROM PRODUCTOS1 WHERE SECCION='DEPORTIVOS'  
2 AND PRECIO BETWEEN 50 AND 100
```

Le damos al botón continuar.



Seleccionaremos Aceptar.

✓ 3 filas afectadas. (La consulta tardó 0,0003 segundos.)

Se han eliminado 3 registros.

Ahora tenemos que eliminar a CLIENTES1 para volver a realizar la consulta y generar de nuevo otra tabla llamada CLIENTES1 que contenga los registros que eliminamos.

```
1 CREATE TABLE CLIENTES1 SELECT * FROM CLIENTES
```

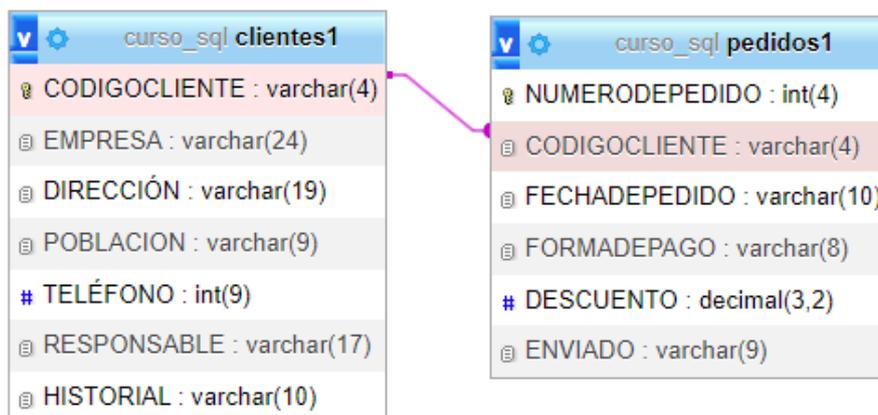
Creamos un índice.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	CODIGOCLIENTE	38	A	No	

En la tabla PEDIDOS1 creamos los correspondientes índices.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	NUMERODEPEDIDO	52	A	No	
Editar Renombrar Eliminar	CODIGOCLIENTE	BTREE	No	No	CODIGOCLIENTE	52	A	Sí	

Ahora vamos a relacionar las dos tablas:



Hay que tener en cuenta con las consultas de predicado que son aquellas que incluimos un predicado DISTINCT o DISTINCTROW.

Queremos saber los clientes que nos han hecho pedidos.

```
1 SELECT EMPRESA FROM CLIENTES1 INNER JOIN PEDIDOS1
2 ON CLIENTES1.CODIGOCLIENTE=PEDIDOS1.CODIGOCLIENTE
```

Le damos a continuar:

EMPRESA	EMPRESA	EMPRESA
BELTRÁN E HIJOS	DEPORTES MORÁN	BAZAR LA FARAONA
BELTRÁN E HIJOS	JUGUETES LA SONRISA	BAZAR LA FARAONA
BELTRÁN E HIJOS	JUGUETES LA SONRISA	
BELTRÁN E HIJOS	LA CURTIDORA	
BELTRÁN E HIJOS	LÍNEA JOVEN	
BELTRÁN E HIJOS	EL PALACIO DE LA MODA	
BELTRÁN E HIJOS	SÁEZ Y CÍA	
BELTRÁN E HIJOS	DEPORTES EL MADRILEÑO	
BELTRÁN E HIJOS	DEPORTES EL MADRILEÑO	
LA MODERNA	FERRETERÍA LA ESCOBA	
LA MODERNA	FERRETERÍA LA ESCOBA	
EXPORTASA	CONFECCIONES HERMINIA	
EXPORTASA	CONFECCIONES HERMINIA	
EXPORTASA	CONFECCIONES HERMINIA	
CONFECCIONES AMPARO	BAZAR EL ARGENTINO	
CONFECCIONES AMPARO	BAZAR EL ARGENTINO	
LA CASA DEL JUGUETE	BAZAR EL ARGENTINO	
ALMACÉN POPULAR	BAZAR EL ARGENTINO	
ALMACÉN POPULAR	BAZAR EL ARGENTINO	
ALMACÉN POPULAR	BAZAR EL ARGENTINO	
ALMACÉN POPULAR	BAZAR EL ARGENTINO	
ALMACÉN POPULAR	LA TIENDA ELEGANTE	
FERRETERÍA EL CLAVO	BAZAR LA FARAONA	
FERNÁNDEZ SL	BAZAR LA FARAONA	
CONFECCIONES ARTÍMEZ	BAZAR LA FARAONA	

La información es correcta pero está repetida, porque los clientes han realizado varios pedidos.

DISTINCT poniendo en un campo que no se repita, le estamos diciendo que muestre un registro por empresa.

```
1 SELECT DISTINCT EMPRESA FROM CLIENTES1 INNER JOIN PEDIDOS1 ON  
2 CLIENTES1.CODIGOCLIENTE=PEDIDOS1.CODIGOCLIENTE
```

Le damos a continuar.

EMPRESA
BELTRÁN E HIJOS
LA MODERNA
EXPORTASA
CONFECCIONES AMPARO
LA CASA DEL JUGUETE
ALMACÉN POPULAR
FERRETERÍA EL CLAVO
FERNÁNDEZ SL
CONFECCIONES ARTÍMEZ
DEPORTES GARCÍA
DEPORTES MORÁN
JUGUETES LA SONRISA
LA CURTIDORA
LÍNEA JOVEN
EL PALACIO DE LA MODA
SÁEZ Y CÍA
DEPORTES EL MADRILEÑO
FERRETERÍA LA ESCOBA
CONFECCIONES HERMINIA
BAZAR EL ARGENTINO
LA TIENDA ELEGANTE
BAZAR LA FARAONA

Ahora los clientes no se repiten.

DISTINCTROW lo que hace no mostrar registros repetidos, en el caso de que en una tabla hubiera 2 o más registros repetidos DISTINCTROW en una consulta mostraría este registro solo una única vez.

En la tabla PRODUCTOS1 hemos agregado un nuevo registro que se repite con uno que ya estaba añadido.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	FALSO	ESPAÑA	NULL

Se encuentra al inicio y al final de la tabla.

```
1 SELECT * FROM PRODUCTOS1 WHERE SECCION='FERRETERÍA'
```

Le damos a continuar.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	FALSO	ESPAÑA	NULL
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23	VERDADERO	USA	NULL
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA	NULL
AR22	FERRETERÍA	MARTILLO	11.3952	2001-09-04	FALSO	ESPAÑA	NULL
AR30	FERRETERÍA	DESTORNILLADOR	9.0584	2002-02-20	VERDADERO	FRANCIA	NULL
AR35	FERRETERÍA	LIMA GRANDE	22.0692	2002-08-10	FALSO	ESPAÑA	NULL
AR36	FERRETERÍA	JUEGO DE BROCAS	15.0950	2002-07-04	VERDADERO	TAIWÁN	NULL
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	FALSO	ESPAÑA	NULL

Observamos como se muestran los dos registros repetidos.

```
1 SELECT DISTINCTROW * FROM PRODUCTOS1 WHERE SECCION='FERRETERÍA'
```

Añadimos el DISTINCTROW y le damos al botón Continuar.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	FALSO	ESPAÑA	NULL
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23	VERDADERO	USA	NULL
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	VERDADERO	ITALIA	NULL
AR22	FERRETERÍA	MARTILLO	11.3952	2001-09-04	FALSO	ESPAÑA	NULL
AR30	FERRETERÍA	DESTORNILLADOR	9.0584	2002-02-20	VERDADERO	FRANCIA	NULL
AR35	FERRETERÍA	LIMA GRANDE	22.0692	2002-08-10	FALSO	ESPAÑA	NULL
AR36	FERRETERÍA	JUEGO DE BROCAS	15.0950	2002-07-04	VERDADERO	TAIWÁN	NULL

Ahora el registro repetido solo se muestra una vez.

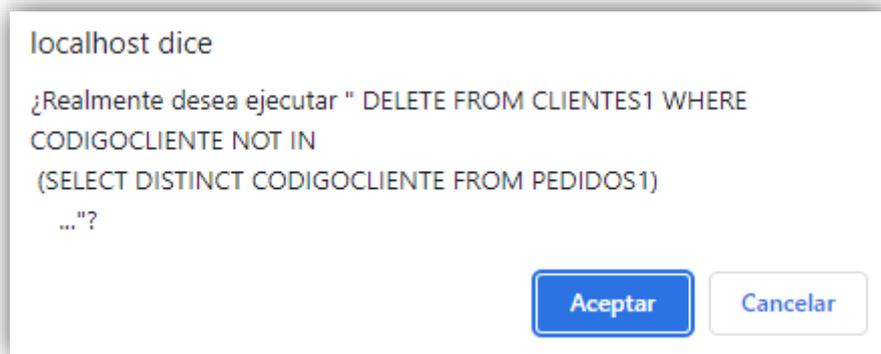
Vamos a realizar una consulta que elimine de la tabla CLIENTES1 los clientes que no han hecho pedidos en la tabla PEDIDOS1.

```
1 DELETE DISTINCTROW CLIENTES1.*, PEDIDOS1.CODIGOCLIENTE
2 FROM CLIENTES1 LEFT JOIN PEDIDOS1 ON CLIENTES1.CODIGOCLIENTE=PEDIDOS1.CODIGOCLIENTE
3 WHERE PEDIDOS1.CODIGOCLIENTE IN NULL
```

La opción explicada por el profesor me da error.

```
1 DELETE FROM CLIENTES1 WHERE CODIGOCLIENTE NOT IN
2 (SELECT DISTINCT CODIGOCLIENTE FROM PEDIDOS1)
```

Vamos a continuar.



Le damos a Aceptar.

✓ 16 filas afectadas. (La consulta tardó 0,0003 segundos.)

Hemos eliminado a 16 clientes que no han realizado pedidos.

Capítulo 14.- Consultas de acción III. Consulta de datos anexados

En este capítulo veremos...

- Datos anexados
 - Insert Into

Tenemos una tabla1 con sus datos y una segunda tabla2 que en su estructura es muy parecida, teniendo el mismo número de campos y mismo tipo de datos en los campos pero los registros son diferentes.

Campo1	Campo2	Campo3	Campo4

Campo1	Campo2	Campo3	Campo4

Una consulta de datos anexados lo que permite es anexar todos los registros o parte de los registros de una tabla en otra.

Campo1	Campo2	Campo3	Campo4

Campo1	Campo2	Campo3	Campo4

Tenemos una tabla que se llama CLIENTES1 y una segunda tabla que se llama CLIENTES_MADRID, en la tabla CLIENTES1 no tenemos ningún cliente que sea de Madrid.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT24	SÁEZ Y CÍA	INFANTA MERCEDES 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ

CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL

Si abrimos la tabla CLIENTES_MADRID.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTÍNEZ
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GÓMEZ
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	912649987	ELÍAS PÉREZ
CT10	FERETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	914354866	MANUEL MENÉNDEZ
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	913299475	ANA JIMÉNEZ
CT19	CONFECCIONES GALÁN	FUENCARRAL 78	MADRID	913859234	JUAN GARCÍA
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	927785235	LAURA CARRASCO
CT26	FERRETERÍA LA ESCOBA	ORENSE 7	MADRID	918459346	JOSÉ GARCÍA
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	912495973	ADRIÁN ÁLVAREZ
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	915483627	ANGEL SANTAMARÍA
CT36	JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID	916872354	PEDRO IGLESIAS
CT39	FERRETERÍA LIMA	VALLECAS 45	MADRID	913532785	LUIS GARCÍA

Esta segunda tabla tiene la misma estructura, mismo número de campos y de tipo iguales, pero los registros son distintos porque esta tabla tiene clientes de la población de Madrid.

Esta situación se da con frecuencia a la hora de trabajar con base de datos o bien porque importamos esta tabla de cualquier fuente de datos, archivo Excel o cualquier compañero que nos proporcione una tabla, pues la cual nos iría muy bien poderla anexar a nuestra tabla.

Es más cómodo manejar la información en una única tabla que tener la información dividida en varias tablas, que es como lo tenemos ahora mismo.

Una consulta de datos anexados nos permite unir la información de una tabla en otra.

Como se hace este tipo de consulta:

```
1 INSERT INTO CLIENTES1
2 SELECT * FROM CLIENTES_MADRID
```

Le damos a continuar.

✓ 12 filas insertadas. (La consulta tardó 0,0011 segundos.)

Ahora vamos a consultar la tabla CLIENTES1.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE	HISTORIAL
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	914456435	ANGEL MARTÍNEZ	NULL
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA	NULL
CT04	EXPORTASA	VALLECAS 34	MADRID	913452378	ELVIRA GÓMEZ	NULL
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ	NULL
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	912649987	ELÍAS PÉREZ	NULL
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ	NULL
CT10	FERETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	914354866	MANUEL MENÉNDEZ	NULL
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO	NULL

CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	913299475	ANA JIMÉNEZ
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT19	CONFECCIONES GALÁN	FUENCARRAL 78	MADRID	913859234	JUAN GARCÍA
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	927785235	LAURA CARRASCO
CT24	SÁEZ Y CÍA	INFANTA MERCEDES 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ
CT26	FERRETERÍA LA ESCOBA	ORENSE 7	MADRID	918459346	JOSÉ GARCÍA
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	912495973	ADRIÁN ÁLVAREZ
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	915483627	ANGEL SANTAMARÍA
CT36	JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID	916872354	PEDRO IGLESIAS
CT39	FERRETERÍA LIMA	VALLECAS 45	MADRID	913532785	LUIS GARCÍA

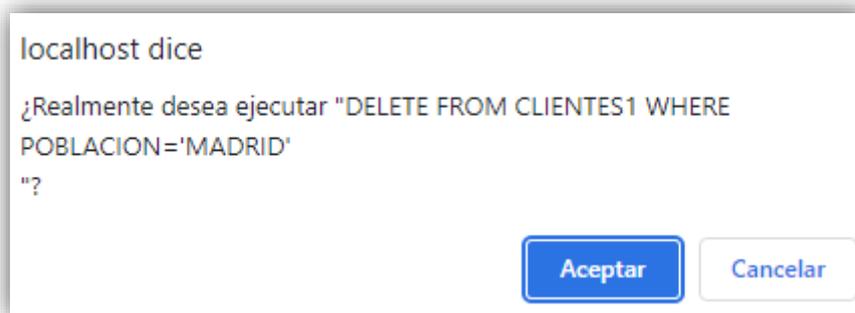
Ya hemos agregado los clientes de Madrid a la tabla CLIENTES1, y además los ordena por el campo clave.

Con respecto de anexas parte de los campos y no todos hay que tener una precaución si son campos clave o campos requeridos.

Vamos a eliminar de nuevo aquellos clientes de son de Madrid de la tabla CLIENTES1.

```
1 DELETE FROM CLIENTES1 WHERE POBLACION='MADRID'
```

Le damos a continuar.



Decimos aceptar.

✓ 12 filas afectadas. (La consulta tardó 0,0014 segundos.)

Ya tenemos la tabla CLIENTES1 sin los clientes de Madrid.

Ahora vamos a volver a anexar los clientes de Madrid a la tabla CLIENTES1, pero no todos los campos, solo CODIGOCLIENTE, EMPRESA, POBLACION y TELEFONO.

```
1 INSERT INTO CLIENTES1 (CODIGOCLIENTE, EMPRESA, POBLACION, TELEFONO)
2 SELECT CODIGOCLIENTE, EMPRESA, POBLACION, TELEFONO FROM CLIENTES_MADRID
```

Le damos a continuar.

✓ 12 filas insertadas. (La consulta tardó 0,0007 segundos.)

Ahora vamos a consultar la tabla CLIENTES1.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT01	BELTRÁN E HIJOS	NULL	MADRID	914456435	NULL
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA
CT04	EXPORTASA	NULL	MADRID	913452378	NULL
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ
CT07	LA CASA DEL JUGUETE	NULL	MADRID	912649987	NULL
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ
CT10	FERRETERÍA EL CLAVO	NULL	MADRID	914354866	NULL
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT14	DEPORTES GARCÍA	NULL	MADRID	913299475	NULL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT19	CONFECCIONES GALÁN	NULL	MADRID	913859234	NULL
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT23	EL PALACIO DE LA MODA	NULL	MADRID	927785235	NULL
CT24	SÁEZ Y CÍA	INFANTA MERCEDES 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ
CT26	FERRETERÍA LA ESCOBA	NULL	MADRID	918459346	NULL
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT30	BAZAR EL ARGENTINO	NULL	MADRID	912495973	NULL
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL
CT34	BAZAR LA FARAONA	NULL	MADRID	915483627	NULL
CT36	JUGUETES EDUCATIVOS SANZ	NULL	MADRID	916872354	NULL
CT39	FERRETERÍA LIMA	NULL	MADRID	913532785	NULL

Podemos ver los campos que no hemos añadido.

Hay que tener la precaución de no manejar campos clave, es decir en este caso en concreto la tabla CLIENTES1 el campo CODIGOCLIENTE es campo clave no lo podemos dejar vacío, ni tampoco se puede repetir, por lo cual hay que tener la precaución que en la tabla de procedencia el

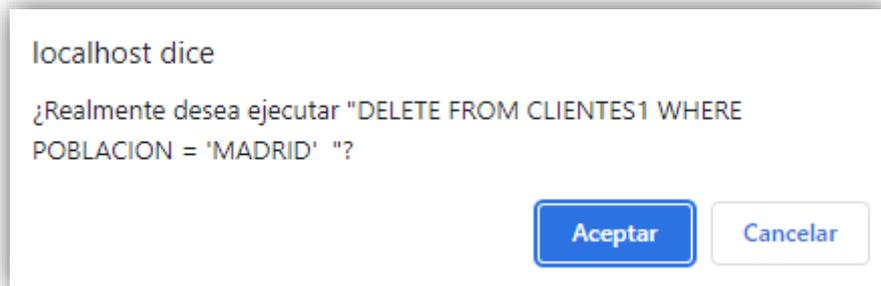
CODIGOCLIENTE no es el mismo que en la tabla de destino, de algún cliente que tuviera el mismo CODIGOCLIENTE.

También hay que tener en cuenta los campos requeridos que en esta tabla no hay, pero si los hubiera estos tendrían que añadirse a la tabla CLIENTES1.

Vamos a realizar una prueba, para ello vamos a eliminar de nuevo los clientes de Madrid de la tabla CLIENTES1.

```
1 DELETE FROM CLIENTES1 WHERE POBLACION = 'MADRID'
```

Le damos a continuar.



Contestamos con Aceptar.

✓ 12 filas afectadas. (La consulta tardó 0,0027 segundos.)

Comprobamos que en la tabla no hay ningún cliente de Madrid.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT24	SÁEZ Y CÍA	INFANTA MERCEDS 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL

Vamos a anexar la tabla CLIENTES_MADRID pero nos insertamos el campo CODIGOCLIENTE este es un campo obligatorio.

```

1 INSERT INTO CLIENTES1 (EMPRESA, DIRECCION, POBLACION)
2 SELECT EMPRESA, DIRECCION, POBLACION FROM CLIENTES_MADRID

```

Le damos a continuar.

Error

consulta SQL: [Copiar](#)

```

INSERT INTO CLIENTES1 (EMPRESA, DIRECCIÓN, POBLACION)
SELECT EMPRESA, DIRECCIÓN, POBLACION FROM CLIENTES_MADRID;

```

MySQL ha dicho:

#1062 - Entrada duplicada '' para la clave 'PRIMARY'

No se ha podido anexar ningún registro.

CODIGOCLIENTE	EMPRESA	DIRECCIÓN	POBLACION	TELÉFONO	RESPONSABLE
CT02	LA MODERNA	LA PALOMA 123	OVIEDO	985323434	JUAN GARCÍA
CT06	CONFECCIONES AMPARO	LOS MOROS 23	GIJÓN	985754332	LUÍS ÁLVAREZ
CT09	ALMACÉN POPULAR	LAS FUENTES 124	BILBAO	942347127	JOSÉ ÁLVAREZ
CT12	FERNÁNDEZ SL	PASEO DEL MAR 45	SANTANDER	942049586	ELISA COLLADO
CT13	CONFECCIONES ARTÍMEZ	GENERAL PERÓN 45	A CORUÑA	981345239	ESTEBAN PASCUAL
CT16	DEPORTES MORÁN	AUTONOMÍA 45	LUGO	982986944	JOSÉ MANZANO
CT18	JUGUETES LA SONRISA	LA BAÑEZA 67	LEÓN	987945368	FAUSTINO PÉREZ
CT20	LA CURTIDORA	OLIVARES 3	MÁLAGA	953756259	MARÍA GÓMEZ
CT21	LÍNEA JOVEN	SIERPES 78	SEVILLA	953452567	ASUNCIÓN SALADO
CT24	SÁEZ Y CÍA	INFANTA MERCEDS 23	SEVILLA	954869234	MANUEL GUERRA
CT25	DEPORTES EL MADRILEÑO	CASTILLA 345	ZARAGOZA	976388934	CARLOS GONZÁLEZ
CT28	CONFECCIONES HERMINIA	CORRIDA 345	GIJÓN	985597315	ABEL GONZÁLEZ
CT31	LA TIENDA ELEGANTE	EL COMENDADOR 67	ZARAGOZA	975694035	JOSÉ PASCUAL

Capítulo 15.- Referencias cruzadas (Access)

En este capítulo veremos ...

- Consultas de referencias cruzadas

Es lo mismo que una tabla dinámica de Excel.



Las tablas dinámicas de Excel tenemos una zona más, arriba a la izquierda que sería la zona de filtro.

En SQL las consultas van a tener esta estructura, de esta forma podemos ver el resultado de las consultas en formato de tabla, pues cuando tenemos muchos datos en una tabla en ocasiones es muy cómodo verlo así.

Para poder construir una consulta de referencias cruzadas tenemos que utilizar instrucciones que ya conocemos pero también hemos de utilizar instrucciones nuevas.

Por ejemplo para poder construir la Zona / Campo de columnas tenemos que utilizar la instrucción PIVOT.

Para poder construir la Zona / Campo de Totales tenemos que utilizar TRANSFORM.

Para construir la zona / Campo filas tenemos que utilizar GROUP BY, que ya conocemos cuando dimos las consultas de agrupación.

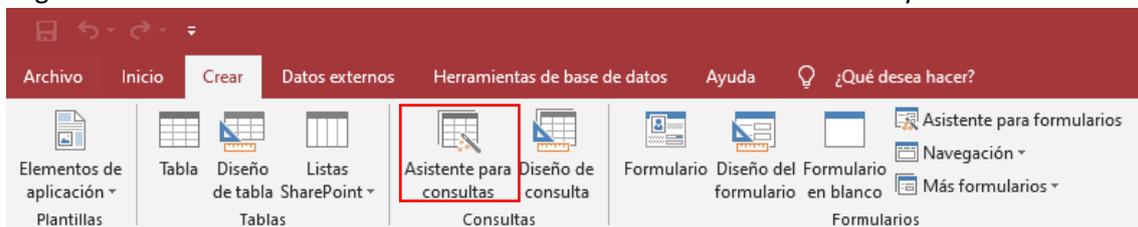
SQL no admite este tipo de consultas, lo tendremos que ver desde Access.

Vamos a ver algunos ejemplos:

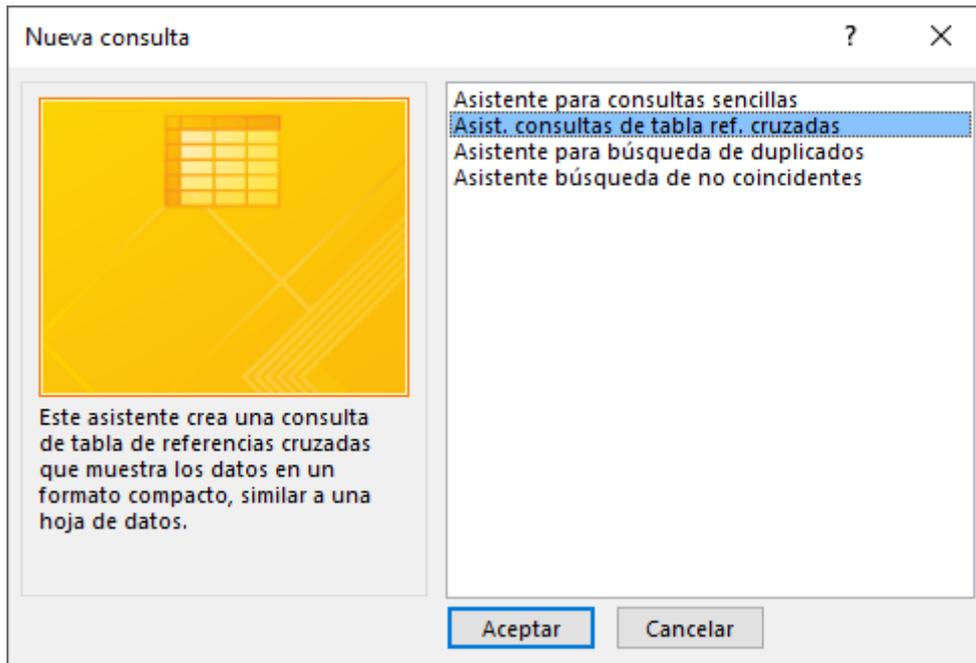
PRODUCTOS							
	CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADC	PAÍSDEORIGEN
+	AR01	FERRETERÍA	DESTORNILLADOR	6,63 €	22/10/2000	<input type="checkbox"/>	ESPAÑA
+	AR02	CONFECCIÓN	TRAJE CABALLERO	284,58 €	11/03/2002	<input checked="" type="checkbox"/>	ITALIA
+	AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159,45 €	26/05/2002	<input checked="" type="checkbox"/>	MARRUECOS
+	AR04	DEPORTES	RAQUETA TENIS	93,47 €	20/03/2000	<input checked="" type="checkbox"/>	USA
+	AR06	DEPORTES	MANCUERNAS	60,00 €	13/09/2000	<input checked="" type="checkbox"/>	USA
+	AR07	CONFECCIÓN	SERRUCHO	30,20 €	23/03/2001	<input checked="" type="checkbox"/>	FRANCIA
+	AR08	JUGUETERÍA	CORREPASILLOS	103,34 €	11/04/2000	<input checked="" type="checkbox"/>	JAPÓN
+	AR09	CONFECCIÓN	PANTALÓN SEÑORA	174,23 €	10/01/2000	<input checked="" type="checkbox"/>	MARRUECOS
+	AR10	JUGUETERÍA	CONSOLA VIDEO	442,54 €	24/09/2002	<input checked="" type="checkbox"/>	USA
+	AR11	CERÁMICA	TUBOS	168,43 €	04/02/2000	<input checked="" type="checkbox"/>	CHINA
+	AR12	FERRETERÍA	LLAVE INGLESA	24,40 €	23/05/2001	<input checked="" type="checkbox"/>	USA
+	AR13	CONFECCIÓN	CAMISA CABALLERO	67,13 €	11/08/2002	<input type="checkbox"/>	ESPAÑA
+	AR14	JUGUETERÍA	TREN ELÉCTRICO	1.505,38 €	03/07/2001	<input checked="" type="checkbox"/>	JAPÓN
+	AR15	CERÁMICA	PLATO DECORATIVO	54,09 €	07/06/2000	<input checked="" type="checkbox"/>	CHINA
+	AR16	FERRETERÍA	ALICATES	6,74 €	17/04/2000	<input checked="" type="checkbox"/>	ITALIA
+	AR17	JUGUETERÍA	MUÑECA ANDADORA	105,06 €	04/01/2001	<input type="checkbox"/>	ESPAÑA
+	AR18	DEPORTES	PISTOLA OLÍMPICA	46,73 €	02/02/2001	<input checked="" type="checkbox"/>	SUECIA
+	AR19	CONFECCIÓN	BLUSA SRA.	101,06 €	18/03/2000	<input checked="" type="checkbox"/>	CHINA
+	AR20	CERÁMICA	JUEGO DE TE	43,27 €	15/01/2001	<input checked="" type="checkbox"/>	CHINA
+	AR21	CERÁMICA	CENICERO	19,75 €	02/07/2001	<input checked="" type="checkbox"/>	JAPÓN
+	AR22	FERRETERÍA	MARTILLO	11,40 €	04/09/2001	<input type="checkbox"/>	ESPAÑA
+	AR23	CONFECCIÓN	CAZADORA PIEL	522,69 €	10/07/2001	<input checked="" type="checkbox"/>	ITALIA
+	AR24	DEPORTES	BALÓN RUGBY	111,64 €	11/11/2000	<input checked="" type="checkbox"/>	USA
+	AR25	DEPORTES	BALÓN BALONCESTO	75,27 €	25/06/2001	<input checked="" type="checkbox"/>	JAPÓN
+	AR26	JUGUETERÍA	FUERTE DE SOLDADOS	143,70 €	25/11/2000	<input checked="" type="checkbox"/>	JAPÓN
+	AR27	CONFECCIÓN	ABRIGO CABALLERO	500.000,00 €	05/04/2002	<input checked="" type="checkbox"/>	ITALIA
+	AR28	DEPORTES	BALÓN FÚTBOL	43,91 €	04/07/2002	<input type="checkbox"/>	ESPAÑA
+	AR29	CONFECCIÓN	ABRIGO SRA	360,07 €	03/05/2001	<input checked="" type="checkbox"/>	MARRUECOS
+	AR30	FERRETERÍA	DESTORNILLADOR	9,06 €	20/02/2002	<input checked="" type="checkbox"/>	FRANCIA
+	AR31	JUGUETERÍA	PISTOLA CON SONIDOS	57,25 €	15/04/2001	<input type="checkbox"/>	ESPAÑA
+	AR32	DEPORTES	CRONÓMETRO	439,18 €	03/01/2002	<input checked="" type="checkbox"/>	USA
+	AR33	CERÁMICA	MACETA	29,04 €	23/02/2000	<input type="checkbox"/>	ESPAÑA
+	AR34	OFICINA	PIE DE LÁMPARA	39,76 €	27/05/2001	<input checked="" type="checkbox"/>	TURQUÍA
+	AR35	FERRETERÍA	LIMA GRANDE	22,07 €	10/08/2002	<input type="checkbox"/>	ESPAÑA
+	AR36	FERRETERÍA	JUEGO DE BROCAS	15,10 €	04/07/2002	<input checked="" type="checkbox"/>	TAIWÁN
+	AR37	CONFECCIÓN	CINTURÓN DE PIEL	4,33 €	12/05/2002	<input type="checkbox"/>	ESPAÑA
+	AR38	DEPORTES	CAÑA DE PESCA	270,00 €	14/02/2000	<input checked="" type="checkbox"/>	USA
+	AR39	CERÁMICA	JARRA CHINA	127,77 €	02/09/2002	<input checked="" type="checkbox"/>	CHINA
+	AR40	DEPORTES	BOTA ALPINISMO	144,00 €	05/05/2002	<input type="checkbox"/>	ESPAÑA
+	AR41	DEPORTES	PALAS DE PING PONG	21,60 €	02/02/2002	<input type="checkbox"/>	ESPAÑA

Imaginaros que quiero hacer una consulta SQL en la cual podemos ver el ARTÍCULO, SECCIÓN y el PRECIO de cada uno de los artículos.

El gestor de Access trae un asistente de hacer la consulta de una forma muy sencilla.



En la pestaña de Crear seleccionaremos Asistente para consultas.



Tenemos el asistente de consultas de tablas de referencias cruzadas.

Le damos a cancelar, ya que lo vamos ha realizar con instrucciones SQL.

Vamos a diseño de consulta y seleccionamos SQL.



La tabla va a tener la siguiente distribución.

<h1>SECCIÓN</h1>	
<h1>ARTICULOS</h1>	<h1>PRECIO</h1>

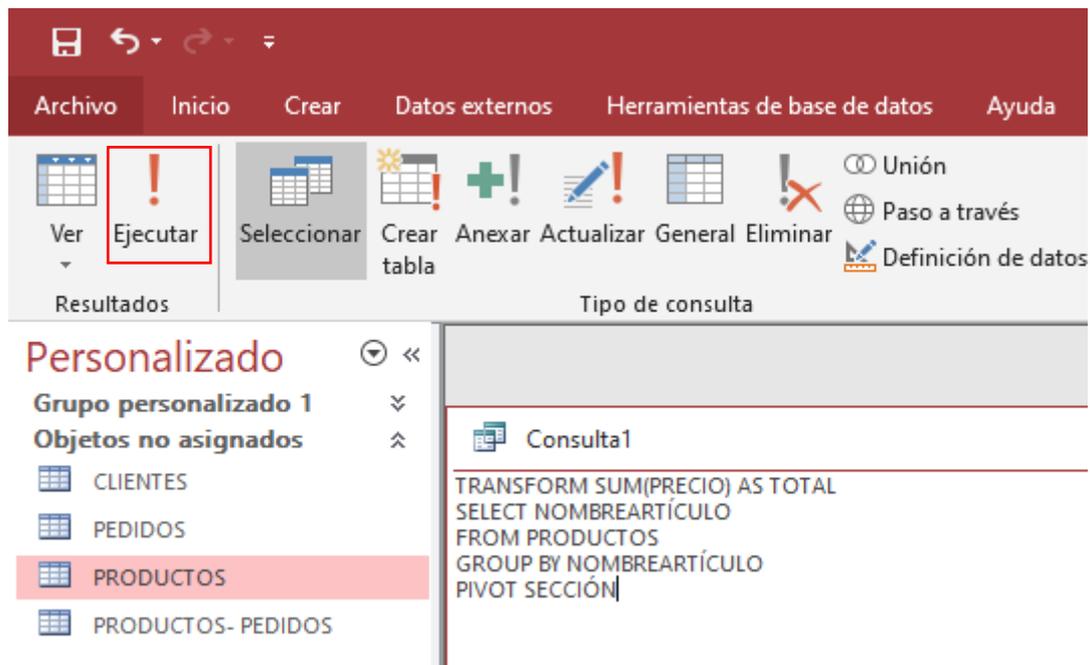
Esta será las instrucción SQL desde Access.

```

Consulta1
-----
TRANSFORM SUM(PRECIO) AS TOTAL
SELECT NOMBREARTÍCULO
FROM PRODUCTOS
GROUP BY NOMBREARTÍCULO
PIVOT SECCIÓN

```

Vamos a ejecutar:



NOMBREARTÍCULO	CERÁMICA	CONFECCIÓ	DEPORTES	FERRETERÍA	JUGUETERÍA	OFICINA
ABRIGO CABALLERO		500.000,00 €				
ABRIGO SRA		360,07 €				
ALICATES				6,74 €		
BALÓN BALONCESTO			75,27 €			
BALÓN FÚTBOL			43,91 €			
BALÓN RUGBY			111,64 €			
BLUSA SRA.		101,06 €				
BOTA ALPINISMO			144,00 €			
CAMISA CABALLERO		67,13 €				
CAÑA DE PESCA			270,00 €			
CAZADORA PIEL		522,69 €				
CENICERO	19,75 €					
CINTURÓN DE PIEL		4,33 €				
COCHE TELEDIRIGIDO					159,45 €	
CONSOLA VIDEO					442,54 €	
CORREPASILLOS					103,34 €	
CRONÓMETRO			439,18 €			
DESTORNILLADOR				15,69 €		
FUERTE DE SOLDADOS					143,70 €	
JARRA CHINA	127,77 €					

En este caso a sumado los dos tipos de destornilladores que hay.

JUEGO DE BROCAS					15,10 €		
JUEGO DE TE	43,27 €						
LIMA GRANDE					22,07 €		
LLAVE INGLESA					24,40 €		
MACETA	29,04 €						
MANCUERNAS				60,00 €			
MARTILLO					11,40 €		
MUÑECA ANDADORA						105,06 €	
PALAS DE PING PONG				21,60 €			
PANTALÓN SEÑORA		174,23 €					
PIE DE LÁMPARA							39,76 €
PISTOLA CON SONIDOS						57,25 €	
PISTOLA OLÍMPICA				46,73 €			
PLATO DECORATIVO	54,09 €						
RAQUETA TENIS				93,47 €			
SERRUCHO		30,20 €					
TRAJE CABALLERO		284,58 €					
TREN ELÉCTRICO						1.505,38 €	
TUBOS	168,43 €						

Podemos cambiar la distribución de la tabla.



Consulta1

```

TRANSFORM SUM(PRECIO) AS TOTAL
SELECT SECCIÓN
FROM PRODUCTOS
GROUP BY SECCIÓN
PIVOT NOMBREARTÍCULO

```

Si ejecutamos:

SECCIÓN	ABRIGO CAE	ABRIGO SRA	ALICATES	BALÓN BALC	BALÓN FÚTE	BALÓN RUG	BLUSA SRA_	BOTA ALPIN
CERÁMICA								
CONFECCIÓN	500.000,00 €	360,07 €					101,06 €	
DEPORTES				75,27 €	43,91 €	111,64 €		144,00 €
FERRETERÍA			6,74 €					
JUGUETERÍA								
OFICINA								

CAMISA CAE	CAÑA DE PE	CAZADORA	CENICERO	CINTURÓN I	COCHE TELE	CONSOLA V	CORREPASIL	CRONOMET
			19,75 €					
67,13 €		522,69 €		4,33 €				
	270,00 €							439,18 €
					159,45 €	442,54 €	103,34 €	

DESTORNILL	FUERTE DE S	JARRA CHIN	JUEGO DE BI	JUEGO DE TE	LIMA GRANI	LLAVE INGLE	MACETA	MANCUERN
		127,77 €		43,27 €			29,04 €	
								60,00 €
15,69 €			15,10 €		22,07 €	24,40 €		
	143,70 €							

MARTILLO	MUÑECA AN	PALAS DE PI	PANTALÓN	PIE DE LÁMF	PISTOLA CO	PISTOLA OLÍ	PLATO DECC	RAQUETA TE
			174,23 €				54,09 €	
		21,60 €				46,73 €		93,47 €
11,40 €								
	105,06 €				57,25 €			
				39,76 €				

SERRUCHO	TRAJE CABA	TREN ELÉCTI	TUBOS
30,20 €	284,58 €		168,43 €
		1.505,38 €	

En este caso a lo ancho es más difícil de consultar.

Queremos consultar el numero unidades por artículo y sección.

```

Consulta referencias cruzadas
TRANSFORM COUNT(PRECIO) AS UNIDADES
SELECT NOMBREARTÍCULO
FROM PRODUCTOS
GROUP BY NOMBREARTÍCULO
PIVOT SECCIÓN;

```

Este será el resultado:

NOMBREARTÍCULO	CERÁMICA	CONFECCIÓ	DEPORTES	FERRETERÍA	JUGUETERÍA	OFICINA
ABRIGO CABALLERO			1			
ABRIGO SRA			1			
ALICATES					1	
BALÓN BALONCESTO				1		
BALÓN FÚTBOL				1		
BALÓN RUGBY				1		
BLUSA SRA.			1			
BOTA ALPINISMO				1		
CAMISA CABALLERO			1			
CAÑA DE PESCA				1		
CAZADORA PIEL			1			
CENICERO		1				
CINTURÓN DE PIEL			1			
COCHE TELEDIRIGIDO						1
CONSOLA VIDEO						1
CORREPASILLOS						1
CRONÓMETRO				1		
DESTORNILLADOR					2	
FUERTE DE SOLDADOS						1
JARRA CHINA		1				
JUEGO DE BROCAS					1	
JUEGO DE TE		1				
LIMA GRANDE					1	
LLAVE INGLESA					1	
MACETA		1				
MANCUERNAS				1		

MARTILLO				1	
MUÑECA ANDADORA					1
PALAS DE PING PONG			1		
PANTALÓN SEÑORA		1			
PIE DE LÁMPARA					1
PISTOLA CON SONIDOS					1
PISTOLA OLÍMPICA			1		
PLATO DECORATIVO	1				
RAQUETA TENIS			1		
SERRUCHO		1			
TRAJE CABALLERO		1			
TREN ELÉCTRICO					1
TUBOS		1			

Sabemos que la tabla CLIENTES y PEDIDOS están relacionadas, de tal forma que podemos ver los pedidos que ha ido haciendo los clientes.

Vamos a realizar una consulta de referencias cruzadas cuantos pedido se han hecho al contado, cuantos pedidos se han hecho aplazado y cuantos con tarjeta.

```

Consulta1
SELECT EMPRESA, POBLACIÓN, FORMADEPAGO
FROM CLIENTES INNER JOIN PEDIDOS ON
CLIENTES.CÓDIGOCLIENTE=PEDIDOS.CÓDIGOCLIENTE

```

Este será el resultado:

EMPRESA	POBLACIÓN	FORMADEPAGO
BELTRÁN E HIJOS	MADRID	CONTADO
BELTRÁN E HIJOS	MADRID	TARJETA
BELTRÁN E HIJOS	MADRID	CONTADO
BELTRÁN E HIJOS	MADRID	CONTADO
BELTRÁN E HIJOS	MADRID	APLAZADO
BELTRÁN E HIJOS	MADRID	TARJETA
BELTRÁN E HIJOS	MADRID	CONTADO
BELTRÁN E HIJOS	MADRID	CONTADO
BELTRÁN E HIJOS	MADRID	CONTADO
LA MODERNA	OVIEDO	CONTADO
LA MODERNA	OVIEDO	CONTADO
EXPORTASA	MADRID	CONTADO
EXPORTASA	MADRID	CONTADO
EXPORTASA	MADRID	TARJETA
CONFECCIONES AMPARO	GIJÓN	CONTADO
CONFECCIONES AMPARO	GIJÓN	CONTADO
LA CASA DEL JUGUETE	MADRID	TARJETA
ALMACÉN POPULAR	BILBAO	APLAZADO
ALMACÉN POPULAR	BILBAO	CONTADO
ALMACÉN POPULAR	BILBAO	APLAZADO
ALMACÉN POPULAR	BILBAO	CONTADO
ALMACÉN POPULAR	BILBAO	APLAZADO

FERRETERÍA EL CLAVO	MADRID	CONTADO
FERNÁNDEZ SL	SANTANDER	CONTADO
CONFECCIONES ARTÍMEZ	A CORUÑA	APLAZADO
DEPORTES GARCÍA	MADRID	APLAZADO
DEPORTES MORÁN	LUGO	CONTADO
JUGUETES LA SONRISA	LEÓN	CONTADO
JUGUETES LA SONRISA	LEÓN	CONTADO
LA CURTIDORA	MÁLAGA	TARJETA
LÍNEA JOVEN	SEVILLA	CONTADO
EL PALACIO DE LA MODA	MADRID	APLAZADO
SÁEZ Y CÍA	SEVILLA	TARJETA
DEPORTES EL MADRILEÑO	ZARAGOZA	CONTADO
DEPORTES EL MADRILEÑO	ZARAGOZA	CONTADO
FERRETERÍA LA ESCOBA	MADRID	TARJETA
FERRETERÍA LA ESCOBA	MADRID	CONTADO
CONFECCIONES HERMINIA	GIJÓN	APLAZADO
CONFECCIONES HERMINIA	GIJÓN	CONTADO
CONFECCIONES HERMINIA	GIJÓN	CONTADO
BAZAR EL ARGENTINO	MADRID	TARJETA
BAZAR EL ARGENTINO	MADRID	TARJETA
BAZAR EL ARGENTINO	MADRID	TARJETA
BAZAR EL ARGENTINO	MADRID	APLAZADO
BAZAR EL ARGENTINO	MADRID	TARJETA
BAZAR EL ARGENTINO	MADRID	TARJETA
LA TIENDA ELEGANTE	ZARAGOZA	CONTADO
BAZAR LA FARAONA	MADRID	CONTADO
BAZAR LA FARAONA	MADRID	APLAZADO
BAZAR LA FARAONA	MADRID	APLAZADO
BAZAR LA FARAONA	MADRID	APLAZADO
BAZAR LA FARAONA	MADRID	CONTADO

Guardamos esta consulta con el nombre de PREVIA.

Ahora vamos a crear una segunda consulta partiendo de la consulta PREVIA.

```

Consulta1
-----
TRANSFORM COUNT(POBLACIÓN) AS TOTAL_FORMAPAGO
SELECT EMPRESA
FROM PREVIA
GROUP BY EMPRESA
PIVOT FORMADEPAGO

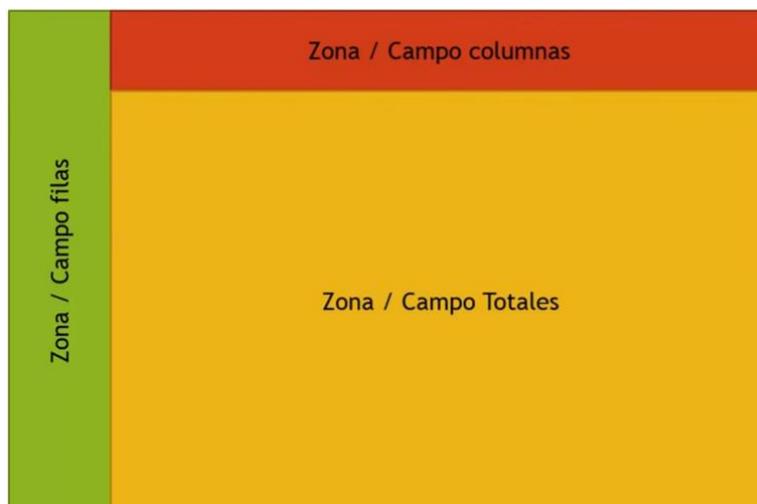
```

Este será el resultado:

Consulta1

EMPRESA	APLAZADO	CONTADO	TARJETA
ALMACÉN POPULAR	3	2	
BAZAR EL ARGENTINO	1		5
BAZAR LA FARAONA	3	2	
BELTRÁN E HIJOS	1	6	2
CONFECCIONES AMPARO		2	
CONFECCIONES ARTÍMEZ	1		
CONFECCIONES HERMINIA	1	2	
DEPORTES EL MADRILEÑO		2	
DEPORTES GARCÍA	1		
DEPORTES MORÁN		1	
EL PALACIO DE LA MODA	1		
EXPORTASA		2	1
FERRETERÍA EL CLAVO		1	
FERNÁNDEZ SL		1	
FERRETERÍA LA ESCOBA		1	1
JUGUETES LA SONRISA		2	
LA CASA DEL JUGUETE			1
LA CURTIDORA			1
LA MODERNA		2	
LA TIENDA ELEGANTE		1	
LÍNEA JOVEN		1	
SÁEZ Y CÍA			1

En una consulta de referencias cruzadas en las que tenemos Columnas, Filas y Total.



En la zona filas se puede incluir más de un campo.

Vamos a revisar una consulta con más de un campo en la zona filas.

TRANSFORM SUM(PRECIO) AS TOTAL
 SELECT NOMBREARTÍCULO, PAÍSDEORIGEN
 FROM PRODUCTOS
 GROUP BY NOMBREARTÍCULO, PAÍSDEORIGEN
 PIVOT SECCIÓN

Este será el resultado:

NOMBREARTÍCULO	PAÍSDEORIGEN	CERÁMICA	CONFECCIÓ	DEPORTES	FERRERERÍA	JUGUETERÍA	OFICINA
ABRIGO CABALLERO	ITALIA		500.000,00 €				
ABRIGO SRA	MARRUECOS		360,07 €				
ALICATES	ITALIA				6,74 €		
BALÓN BALONCESTO	JAPÓN			75,27 €			
BALÓN FÚTBOL	ESPAÑA			43,91 €			
BALÓN RUGBY	USA			111,64 €			
BLUSA SRA.	CHINA		101,06 €				
BOTA ALPINISMO	ESPAÑA			144,00 €			
CAMISA CABALLERO	ESPAÑA		67,13 €				
CAÑA DE PESCA	USA			270,00 €			
CAZADORA PIEL	ITALIA		522,69 €				
CENICERO	JAPÓN	19,75 €					
CINTURÓN DE PIEL	ESPAÑA		4,33 €				
COCHE TELEDIRIGIDO	MARRUECOS					159,45 €	
CONSOLA VIDEO	USA					442,54 €	
CORREPASILLOS	JAPÓN					103,34 €	
CRONÓMETRO	USA			439,18 €			
DESTORNILLADOR	ESPAÑA				6,63 €		
DESTORNILLADOR	FRANCIA				9,06 €		
FUERTE DE SOLDADOS	JAPÓN					143,70 €	
JARRA CHINA	CHINA	127,77 €					
JUEGO DE BROCAS	TAIWÁN				15,10 €		
JUEGO DE TE	CHINA	43,27 €					
LIMA GRANDE	ESPAÑA				22,07 €		
LLAVE INGLESA	USA				24,40 €		
MACETA	ESPAÑA	29,04 €					
MANCUERNAS	USA			60,00 €			
MARTILLO	ESPAÑA				11,40 €		
MUÑECA ANDADORA	ESPAÑA					105,06 €	
PALAS DE PING PONG	ESPAÑA			21,60 €			
PANTALÓN SEÑORA	MARRUECOS		174,23 €				
PIE DE LÁMPARA	TURQUÍA						39,76 €
PISTOLA CON SONIDOS	ESPAÑA					57,25 €	
PISTOLA OLÍMPICA	SUECIA			46,73 €			
PLATO DECORATIVO	CHINA	54,09 €					
RAQUETA TENIS	USA			93,47 €			
SERRUCHO	FRANCIA		30,20 €				
TRAJE CABALLERO	ITALIA		284,58 €				
TREN ELÉCTRICO	JAPÓN					1.505,38 €	
TUBOS	CHINA	168,43 €					

Ahora si aparecen dos destornilladores porque son de distinto país.

Referencias cruzadas

- Realiza una consulta de referencias cruzadas que muestre cuántos artículos hay en la tabla de productos por cada año. El resultado obtenido sea el siguiente:

PAÍSDEORIGEN	TOTAL	2000	2001	2002	PRO
CHINA	5	3	1	1	
ESPAÑA	11	2	3	6	
FRANCIA	2		1	1	
ITALIA	4	1	1	2	
JAPÓN	5	2	3		
MARRUECOS	3	1	1	1	
SUECIA	1		1		
TAIWÁN	1			1	
TURQUÍA	1		1		
USA	7	4	1	2	

Ayuda: Para agrupar por año utiliza la función Format: `Format(FECHA,"yyyy")`

- Realiza una consulta de referencias cruzadas que muestre cuántos artículos importados y no importados han sido enviados, y cuántos importados y no importados hay sin enviar. El resultado sea el siguiente:

IMPORTADO	-1	0	
<input checked="" type="checkbox"/>	5	12	
<input type="checkbox"/>		1	

Solución:

- Consulta 1:**

PAÍSDEORIGEN	TOTAL	2000	2001	2002	PRO
CHINA	5	3	1	1	
ESPAÑA	11	2	3	6	
FRANCIA	2		1	1	
ITALIA	4	1	1	2	
JAPÓN	5	2	3		
MARRUECOS	3	1	1	1	
SUECIA	1		1		
TAIWÁN	1			1	
TURQUÍA	1		1		
USA	7	4	1	2	

```

TRANSFORM COUNT(CÓDIGOARTÍCULO) AS CuentaDeCÓDIGOARTÍCULO
SELECT PAÍSDEORIGEN, Count(CÓDIGOARTÍCULO) AS TOTAL
FROM PRODUCTOS
WHERE Format([FECHA],"yyyy") <> 2015
GROUP BY PAÍSDEORIGEN
PIVOT Format(FECHA,"yyyy");
    
```

- Consulta 2:**

IMPORTADO	-1	0
<input checked="" type="checkbox"/>	5	12
<input type="checkbox"/>		1

Para realizar esta consulta, primero tenemos que realizar una consulta previa con los campos necesarios: CódigoArtículo de la tabla de Productos, Importado y Enviado de la tabla de Pedidos. Estas dos tablas están relacionadas con una relación de varios a varios utilizando como tabla de

unión la tabla de ProductosPedidos, por lo que tenemos que agregar esta última a la consulta. La consulta previa sería la siguiente:

```
PREVIA_ENVIADOC
SELECT PRODUCTOS.CÓDIGOARTÍCULO,IMPORTADO, ENVIADO
FROM PRODUCTOS INNER JOIN
(PEDIDOS INNER JOIN PRODUCTOSPEDIDOS ON PEDIDOS.NÚMERODEPEDIDO = PRODUCTOSPEDIDOS.NÚMERODEPEDIDO)
ON PRODUCTOS.CÓDIGOARTÍCULO = PRODUCTOSPEDIDOS.CÓDIGOARTÍCULO;
```

Una vez guardada esta consulta, realizamos la consulta de referencias cruzadas basándonos en la consulta creada:

```
TRANSFORM COUNT(CÓDIGOARTÍCULO) AS CuentaDeCÓDIGOARTÍCULO
SELECT IMPORTADO
FROM PREVIA_ENVIADOS_IMPORTADOS
GROUP BY IMPORTADO
PIVOT ENVIADO;
```

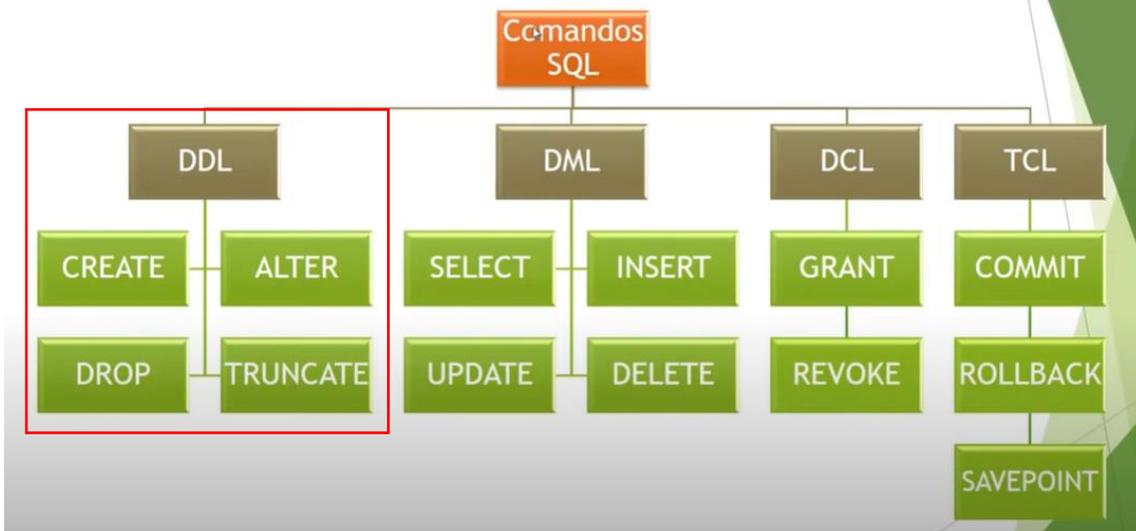
Capítulo 16.- Creación de tablas

En este capítulo veremos...

- Comandos DDL. Data Definition Language. Definición de datos.

Estos comandos nos va a permitir crear tablas, eliminar tablas, crear nuevos campos dentro de una tabla, eliminar campos, modificar las propiedades y características de los campos de las tablas, etc.

Grupos de comandos



CREATE: Permite crear nuevas base de datos y nuevas tablas.

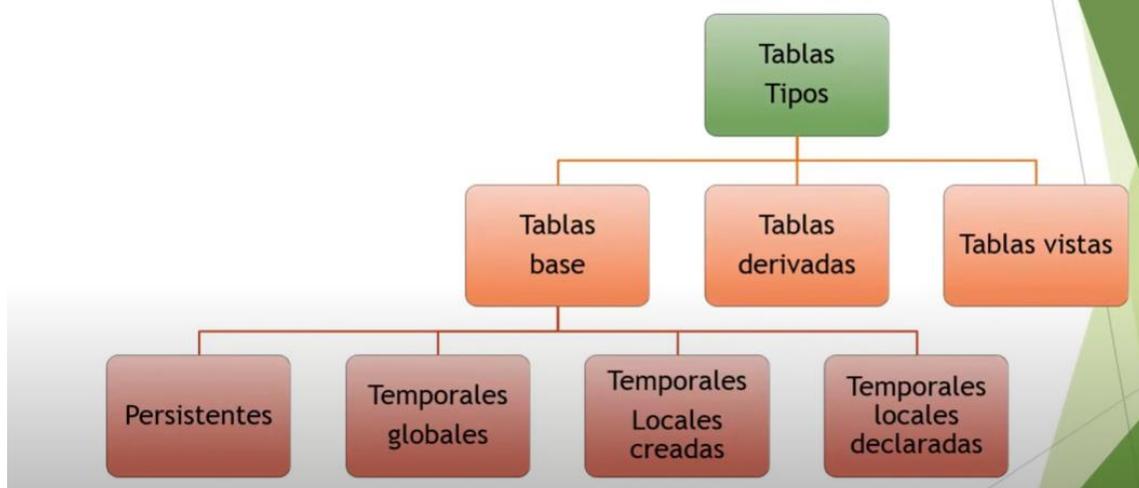
ALTER: Permite modificar las propiedades, los campos de estas nuevas tablas.

DROP: Permite eliminar base de datos y tablas.

TRUNCATE: Permite borrar todas las filas que tenemos en una tabla sin borrar la tabla.

Vamos a empezar por el comando CREATE.

Create. Creación de tablas



Podemos decir que hay tres tipos de tablas, base, derivadas y vistas.

Las tablas base son las que la mayoría conocemos las bases de datos que vamos a manejar en un mayor número de ocasiones.

Dentro de las tablas bases, tenemos las Persistentes, Temporales globales, Temporales Locales creadas y Temporales locales declaradas.

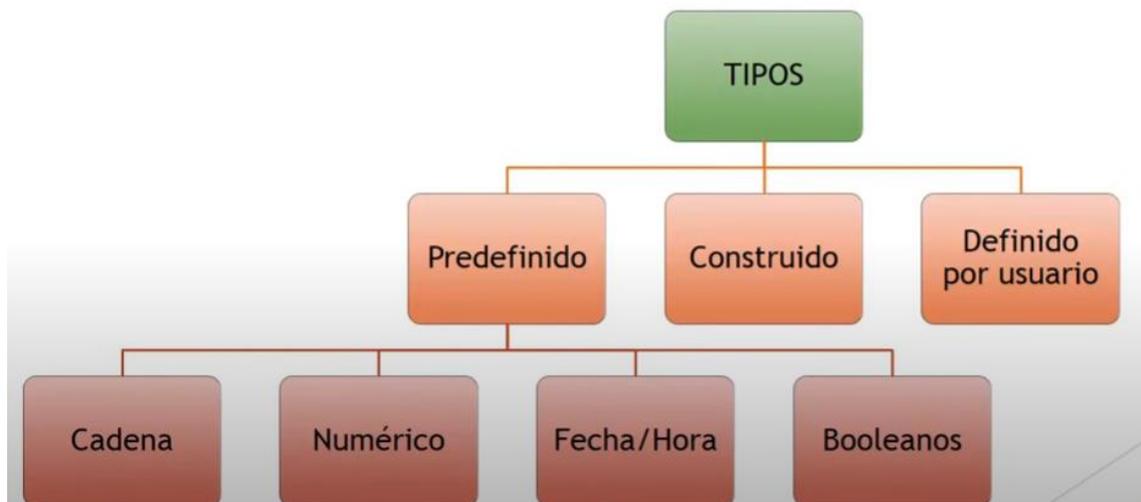
Las tablas temporales son aquellas que se crean en el momento de ejecutar una secuencia SQL esta se puede crear para generar un informe y luego se borrará.

Vamos a trabajar las tablas bases y de estas las Persistentes.

CREATE. Sintaxis.

- CREATE TABLE <NOMBRE DE LA TABLA> (CAMPO1 TIPO_DATO, CAMPO2 TIPO_DATO, CAMPO3 TIPO_DATO...)

TIPOS DE DATOS



Los tipos de datos Predefinidos vienen con el lenguaje estándar SQL pueden ser de tipo Cadena, Numérico, Fecha/Hora y Booleanos. Y depende del gestor de base de datos la sintaxis de los sufixos serán diferentes.

TIPOS DE DATOS SQL			
BYNARY	VARBINARY BINARY VARYING BIT VARYING	1 byte por carácter	Se puede almacenar cualquier valor tipo de datos en un campo de este tipo. Los datos no se traducen (por ejemplo, a texto). La forma en que se introducen los datos en un campo binario indica cómo aparecerán al mostrarlos.
BIT	BOOLEAN LOGICAL LOGICALI	1 byte	Valores Sí o NO, y campos que contienen solamente uno de los dos valores.

	YESNO		
TINYINT	INTERGERI BYTE	1 byte	Número entero entre 0 y 255.
COUNTER	AUTOINCREMENT		Utilizado para campos contadores cuyo valor se incrementa automáticamente al crear un nuevo registro.
MONEY	CURRENCY	8 bytes	Un número entero comprendido entre -922.337.203.685.477, 5808 y 992.337.203.685.477, 5807.
DATETIME	DATE TIME	8 bytes	Un valor de fecha u hora entre los años 100 y 9999.
UNIQUEIDENTIFIER	GUID	128 bits	Un número de identificación único utilizado con llamadas a procedimientos remotos.
DECIMAL	NUMERIC DEC	17 bytes	Un tipo de dato numérico exacto con valores comprendidos entre 1028 -1 y -1028-1. Puede definir la precisión (1-28) y la escala (0 – precisión definida). La precisión y la escala predeterminadas son 18 y 0, respectivamente.
REAL	SINGLE FLOAT4 IEEE SINGLE	4 bytes	Un valor de coma flotante de precisión simple con un intervalo comprendido entre - 3,402823E38 y -1,401298E-45 para valores negativos, y desde 1,401298E-45 a 3,402823E38 para valores positivos, y 0.
FLOAT	DOUBLE FLOAT8 IEEE DOUBLE NUMBER	8 bytes	Un valor de coma flotante de precisión doble con un intervalo comprendido entre 1,79769313486232E08 y -4,94065645841247E-324 para valores negativos, y desde 4,94065645841247E-324 a 1,79769313486232E308 para valores positivos, y 0.
SMALLINT	SHORT INTEGER2	2 bytes	Entero corto entre -32.768 y 32.767.
INTEGER	LONG INT INTEGER4	4 bytes	Entero largo entre -2.147.483.648 y 2.147.483.647.
IMAGE	LONG BINARY GENERAL OLE OBJECT	Lo que se requiera	De cero hasta un máximo de 2.14 gigabytes.
TEXT	LONG TEXT LONG CHAR MEMO NOTE NTEXT	2 bytes por carácter	De cero a 255 caracteres.

CHAR	TEXT(n) ALPHANUMERIC CHARACTER STRING VARCHAR CHARACTER VARYING NCHAR NATIONAL CHARACTER NATIONAL CHAR NATIONAL CHARACTER VARYING NATIONAL CHAR VARYING	2 bytes por ca- rácter.	
------	--	-------------------------------	--

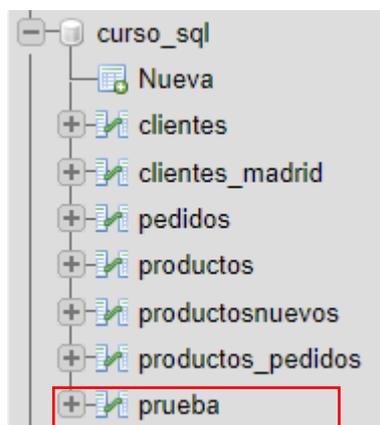
Para más información seleccionar [aquí](#).

```
1 CREATE TABLE PRUEBA (NOMBRE VARCHAR(20))
```

Creamos una tabla con el nombre de PRUEBA que contiene un campo llamado NOMBRE de tipo texto con una longitud de 20 caracteres.

Le damos al botón continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)



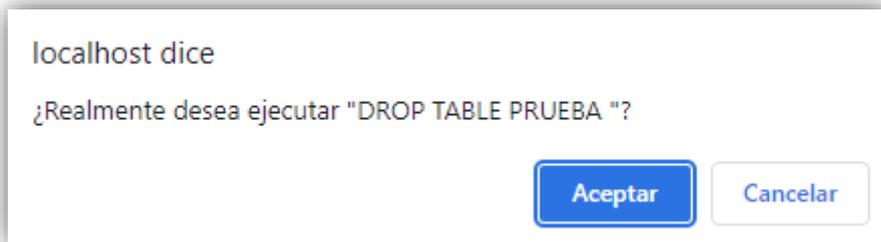
Ya hemos creado la tabla, vamos a ver su estructura.

<input type="checkbox"/>	1	NOMBRE	varchar(20)	utf8mb4_general_ci	Sí	NULL
--------------------------	---	--------	-------------	--------------------	----	------

Vamos a eliminar la tabla PRUEBA desde SQL.

```
1 DROP TABLE PRUEBA
```

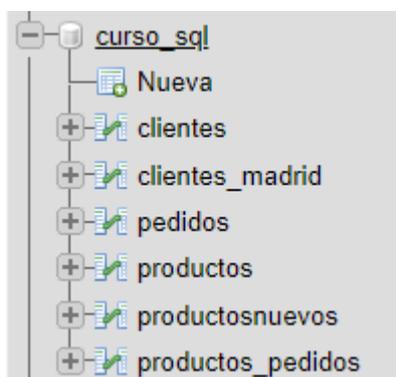
Le damos a continuar.



Le daremos a Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Si miramos la base de datos:



Ya hemos eliminado la tabla PRUEBA.

Si sigues viendo la tabal prueba tendrás que actualizar.



Para que se refresquen los datos de la base de datos.

Vamos a crear otra tabla agregando un campo de más y que sea de diferente tipo.

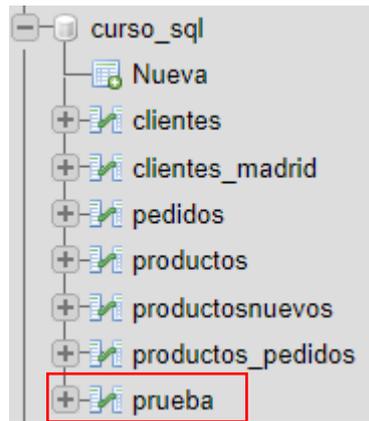
```
1 CREATE TABLE PRUEBA (NOMBRE VARCHAR(20), APELLIDO VARCHAR(20),  
2 EDAD TINYINT, FECHA_NACIMIENTO DATE, CARNET BOOLEAN)
```

Creamos una tabla llamada PRUEBA que tendrá los siguientes campos NOMBRE de tipo texto de 20, APELLIDOS de tipo texto de 20, EDAT de tipo TINYINT, FECHA_NACIMIENTO de tipo DATE y CARNET de tipo BOOLEAN.

Le damos al botón Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la base de datos:



La seleccionamos y a continuación la pestaña estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	1 NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/>	2 APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/>	3 EDAD	tinyint(4)			Sí	NULL
<input type="checkbox"/>	4 FECHA_NACIMIENTO	date			Sí	NULL
<input type="checkbox"/>	5 CARNET	tinyint(1)		Booleano	Sí	NULL

Como podemos declarar un campo que sea de autonumérico con INT AUTO_INCREMENT.

Eliminamos de nuevo la tabla de PRUEBA.

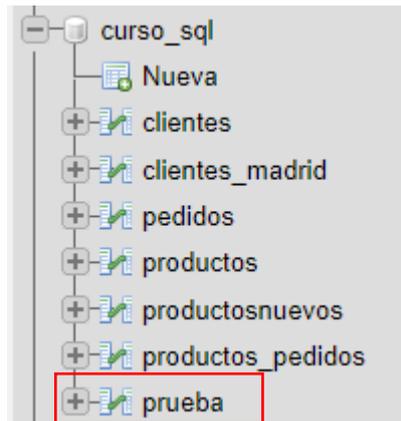
```
1 DROP TABLE PRUEBA
```

```
1 CREATE TABLE PRUEBA (ID_ALUMNO INT AUTO_INCREMENT, NOMBRE VARCHAR(20),
2 APELLIDO VARCHAR(20), EDAD TINYINT,
3 FECHA_NACIMIENTO DATE, CARNET BOOLEAN,
4 PRIMARY KEY (ID_ALUMNO))
```

Creamos una tabla llamada PRUEBA con los siguientes campos ID_ALUMNOS de tipo AUTO-IN-CREMENT, NOMBRE de texto de 20, APELLIDO de texto de 20, EDAD de tipo TINYINT, FECHA_NA-CIMIENTO de tipo DATE, CARNET de tipo BOOLEAN, además al utilizar un campo de tipo AUTO_INCREMENT estamos obligados a especificar el campo clave en ID_ALUMNO.

Le damos a Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)



Ya tenemos de nuevo la tabla PREUBA, la seleccionamos y nos vamos a la pestaña Estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo
1	ID_ALUMNO 	int(11)			No
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí
4	EDAD	tinyint(4)			Sí
5	FECHA_NACIMIENTO	date			Sí
6	CARNET	tinyint(1)			Sí

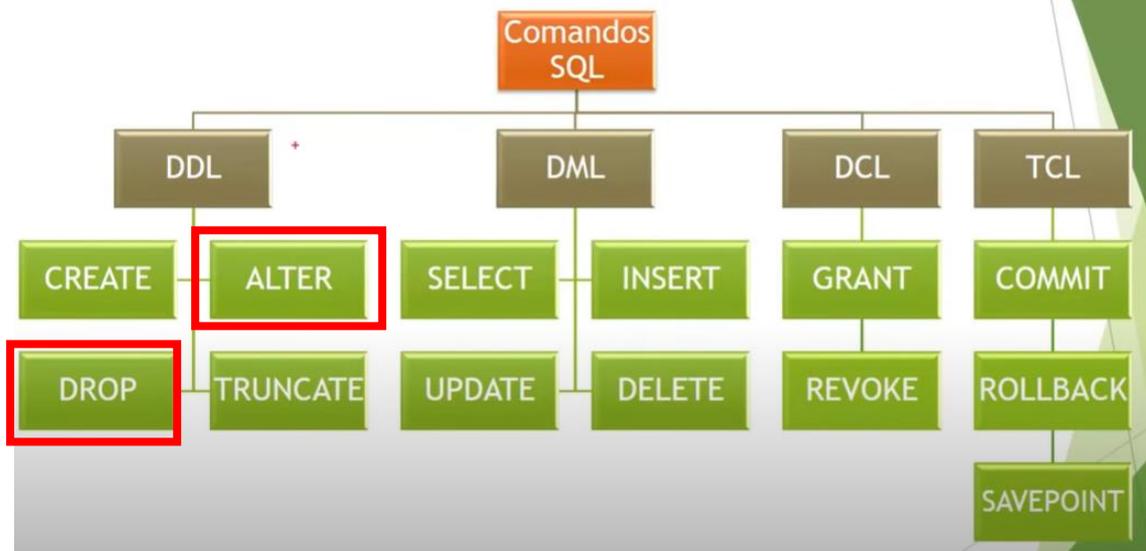
El campo clave se especifica con una llave.

Capítulo 17.- Agregar, eliminar y modificar campos

En este capítulo veremos...

- Agregar, modificar y eliminar campos en una tabla.
- Valores por defecto de los campos de una tabla

Grupos de comandos



ALTER: Con modificación, agregar campos dentro de una tabla.

DROP: Para todo lo relacionado con eliminación de campo como características de un campo de una tabla.

Vamos a comenzar como agregar un campo a una tabla que tuviéramos creada anteriormente.

Si os recordáis tenemos una tabla de CLIENTES_MADRID.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CÓDIGO CLIENTE	varchar(4)	utf8_general_ci		Sí	NULL
2	EMPRESA	varchar(24)	utf8_general_ci		Sí	NULL
3	DIRECCIÓN	varchar(19)	utf8_general_ci		Sí	NULL
4	POBLACIÓN	varchar(9)	utf8_general_ci		Sí	NULL
5	TELÉFONO	varchar(16)	utf8_general_ci		Sí	NULL
6	RESPONSABLE	varchar(17)	utf8_general_ci		Sí	NULL
7	HISTORIAL	varchar(10)	utf8_general_ci		Sí	NULL

Imaginaros que desde el código SQL tenemos que agregar un campo más, con sus características y tipo de datos.

Queremos agregar un campo FECHA_BAJA que sea de tipo Fecha/Hora.

```
1 ALTER TABLE CLIENTES_MADRID ADD COLUMN FECHA_BAJA DATE
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Ahora vamos a ver la estructura de la tabla CLIENTES_MADRID.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CÓDIGO CLIENTE	varchar(4)	utf8_general_ci		Sí	NULL
2	EMPRESA	varchar(24)	utf8_general_ci		Sí	NULL
3	DIRECCIÓN	varchar(19)	utf8_general_ci		Sí	NULL
4	POBLACIÓN	varchar(9)	utf8_general_ci		Sí	NULL
5	TELÉFONO	varchar(16)	utf8_general_ci		Sí	NULL
6	RESPONSABLE	varchar(17)	utf8_general_ci		Sí	NULL
7	HISTORIAL	varchar(10)	utf8_general_ci		Sí	NULL
8	FECHA_BAJA	date			Sí	NULL

Ya tenemos el nuevo campo.

Para el siguiente ejemplo vamos a trabajar con la base de datos PRUEBA.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL

Queremos agregar un campo llamado POBLACION y que sea de tipo texto.

```
1 ALTER TABLE PRUEBA ADD COLUMN POBLACION VARCHAR(20)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Ahora vamos a ver la estructura de la tabla PRUEBA.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL
7	POBLACION	varchar(20)	utf8mb4_general_ci		Sí	NULL

Ya tenemos el nuevo campo.

Si lo que queremos es eliminar el campo de una tabla utilizaremos el comando DROP, lo que hace es eliminar un campo o características de un campo.

Vamos a eliminar el campo POBLACION.

```
1 ALTER TABLE PRUEBA DROP COLUMN POBLACION
```

Le damos a continuar.



Le decimos Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

Ahora vamos a ver la estructura de la tabla PRUEBA.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL

Como verás el campo POBLACION ya se ha eliminado.

Ahora de la tabla CLIENTES_MADRID vamos a eliminar el campo FECHA_BAJA.

```
1 ALTER TABLE CLIENTES_MADRID DROP COLUMN FECHA_BAJA
```

Le damos a continuar.



Le damos a Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura de CLIENTES_MADRID.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CÓDIGO CLIENTE	varchar(4)	utf8_general_ci		Sí	NULL
2	EMPRESA	varchar(24)	utf8_general_ci		Sí	NULL
3	DIRECCIÓN	varchar(19)	utf8_general_ci		Sí	NULL

4	POBLACIÓN	varchar(9)	utf8_general_ci	Sí	NULL
5	TELÉFONO	varchar(16)	utf8_general_ci	Sí	NULL
6	RESPONSABLE	varchar(17)	utf8_general_ci	Sí	NULL
7	HISTORIAL	varchar(10)	utf8_general_ci	Sí	NULL

El campo FECHA_BAJA ya no está en la tabla.

Imaginaros que queremos modificar alguna de las características de un campo, por ejemplo el tipo de dato, para ello vamos a agregar en primer lugar a la tabla CLIENTES_MADRID de nuevo el campo FECHA_BAJA.

```
1 ALTER TABLE CLIENTES_MADRID ADD COLUMN FECHA_BAJA VARCHAR(10)
```

Si te das cuenta hemos cometido un error a posta para poder modificarlo le hemos dicho que FECHA_BAJA es de tipo VARCHAR(10).

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver su estructura:

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CÓDIGO CLIENTE	varchar(4)	utf8_general_ci		Sí	NULL
2	EMPRESA	varchar(24)	utf8_general_ci		Sí	NULL
3	DIRECCIÓN	varchar(19)	utf8_general_ci		Sí	NULL
4	POBLACIÓN	varchar(9)	utf8_general_ci		Sí	NULL
5	TELÉFONO	varchar(16)	utf8_general_ci		Sí	NULL
6	RESPONSABLE	varchar(17)	utf8_general_ci		Sí	NULL
7	HISTORIAL	varchar(10)	utf8_general_ci		Sí	NULL
8	FECHA_BAJA	varchar(10)	utf8mb4_general_ci		Sí	NULL

¿Cómo podremos modificar el tipo de dato que está en texto a fecha?

```
1 ALTER TABLE CLIENTES_MADRID MODIFY FECHA_BAJA DATE
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CÓDIGO CLIENTE	varchar(4)	utf8_general_ci		Sí	NULL
2	EMPRESA	varchar(24)	utf8_general_ci		Sí	NULL
3	DIRECCIÓN	varchar(19)	utf8_general_ci		Sí	NULL
4	POBLACIÓN	varchar(9)	utf8_general_ci		Sí	NULL
5	TELÉFONO	varchar(16)	utf8_general_ci		Sí	NULL
6	RESPONSABLE	varchar(17)	utf8_general_ci		Sí	NULL
7	HISTORIAL	varchar(10)	utf8_general_ci		Sí	NULL
8	FECHA_BAJA	date			Sí	NULL

Ya hemos modificado el tipo de campo para FECHA_BAJA.

Ahora vamos a ver como establecer el valor por defecto de un campo, para ello vamos a utilizar una cláusula SET DEFAULT, que no es admitida por todos los gestores de base de datos.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL

La tabla PRUEBA si nos fijamos en la columna “Predeterminado” en todos aparece NULL, esto significa que no hay ningún campo con un valor predeterminando.

¿Cómo podemos hacer para que en campo que por defecto tiene NULL tenga otro valor por defecto?.

Vamos a agregar un nuevo campo que será LUGAR_NACIMIENTO y que por defecto aparezca el valor predeterminado “DESCONOCIDO”.

```
1 ALTER TABLE PRUEBA ADD COLUMN LUGAR_NACIMIENTO VARCHAR(20)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL
7	LUGAR_NACIMIENTO	varchar(20)	utf8mb4_general_ci		Sí	NULL

Ahora nos queda cambiar el valor predeterminado de LUGAR_NACIMIENTO.

```
1 ALTER TABLE PRUEBA ALTER COLUMN LUGAR_NACIMIENTO SET DEFAULT 'DESCONOCIDO'
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a volver a ver la estructura de PRUEBA.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL

5	FECHA_NACIMIENTO	date		Sí	NULL
6	CARNET	tinyint(1)		Sí	NULL
7	LUGAR_NACIMIENTO	varchar(20)	utf8mb4_general_ci	Sí	DESCONOCIDO

El campo LUGAR_NACIMIENTO ya tiene el valor Predeterminado a DESCONOCIDO.

Si queremos eliminar el valor Predeterminado utilizaremos:

```
1 ALTER TABLE PRUEBA ALTER COLUMN LUGAR_NACIMIENTO DROP DEFAULT
```

Si miramos la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ID_ALUMNO 	int(11)			No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDO	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL
5	FECHA_NACIMIENTO	date			Sí	NULL
6	CARNET	tinyint(1)			Sí	NULL
7	LUGAR_NACIMIENTO	varchar(20)	utf8mb4_general_ci		Sí	NULL

Ya tiene el valor predeterminado como NULL.

Capítulo 18.- Índices

Los índices nos permiten buscar datos en una base de datos con mucha más rapidez, si tu crear un índice en una tabla las búsquedas o consultas que realices en dicha tabla se van a realizar con mayor rapidez. Esto lo vas a notar si tu tabla tiene miles de registros.

Que problema hay con los índices que no forman parte del estándar de SQL, esto quiere decir que cada gestor de base de datos va a tener su propia sintaxis para crear y modificar estos índices.

¿Qué sintaxis vamos a ver en este capítulo? El gestor base de datos MySQL.

En este capítulo veremos...

- Creación de índices:
 - Índices de clave primaria
 - Índices ordinarios
 - Índices únicos
 - Índices compuestos



Los índices de CLAVE PRIMARIA no permiten valores nulos y tampoco permiten valores repetidos.

Los índices ORDINARIOS si que permite valores nulos y demás permiten duplicados.

Los índices ÚNICOS se puede dejar vacío el campo pero no permite duplicados.

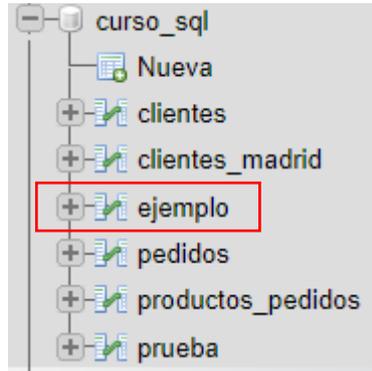
Los índices COMPUESTOS permiten valores nulos pero estos índices están formados por varias columnas.

Lo primero que vamos a hacer es crear una tabla de la manera más sencilla.

```
1 CREATE TABLE EJEMPLO (DNI VARCHAR(15), NOMBRE VARCHAR(20),  
2 APELLIDOS VARCHAR(20) , EDAD TINYINT)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

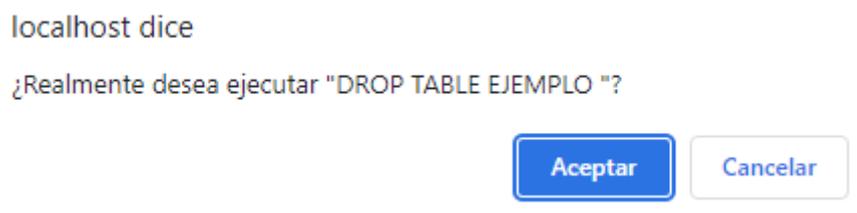


Esta es su estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		Sí	NULL
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL

No hay ningún campo clave, vamos a eliminarla.

```
1 DROP TABLE EJEMPLO
```



✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Contestamos Aceptar.

Vamos a crear la tabla pero añadiendo el índice al final de toda la serie de comandos.

```
1 CREATE TABLE EJEMPLO (DNI VARCHAR(15), NOMBRE VARCHAR(20),  
2 APELLIDOS VARCHAR(20) , EDAD TINYINT,  
3 PRIMARY KEY(DNI))
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

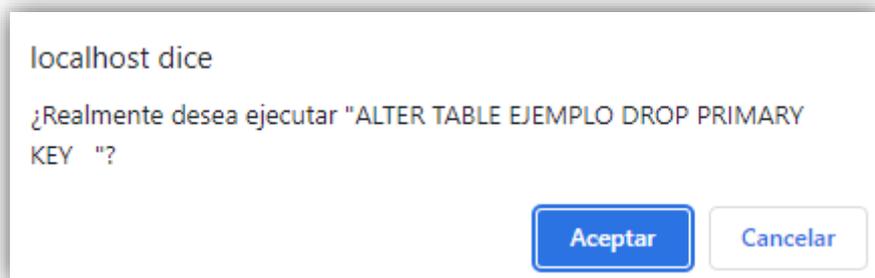
Se ha creado la tabla EJEMPLO, vamos a ver su estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI 	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL

Contiene la llave, esto significa que DNI es el campo clave primaria.

```
1 ALTER TABLE EJEMPLO DROP PRIMARY KEY
```

Le damos a continuar.



Le decimos Aceptar.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL

Ahora el DNI ya no tiene la clave primaria.

Ahora vamos a ver como podemos crear una clave cuando la tabla ya está creada, es decir a posteriori.

```
1 ALTER TABLE EJEMPLO ADD PRIMARY KEY(DNI)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Miramos la estructura de EJEMPLO.

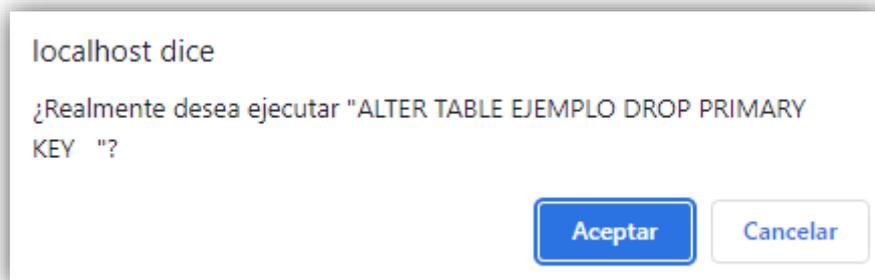
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI 	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		Sí	NULL
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		Sí	NULL
4	EDAD	tinyint(4)			Sí	NULL

El DNI vuelve a tener la clave primaria, en este caso la hemos creado a posteriori.

Las claves primarias pueden estar formadas por un campo o por varios campos,

Vamos de nuevo a quitar la clave primaria del DNI.

```
1 ALTER TABLE EJEMPLO DROP PRIMARY KEY
```



Contestamos Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0006 segundos.)

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna

2	NOMBRE	varchar(20)	utf8mb4_general_ci	Sí	NULL
3	APELLIDOS	varchar(20)	utf8mb4_general_ci	Sí	NULL
4	EDAD	tinyint(4)		Sí	NULL

El campo DNI ya no tiene la clave primaria.

Ahora queremos que la clave principal conste de dos campos, por el NOMBRE y el APELLIDO.

```
1 ALTER TABLE EJEMPLO ADD PRIMARY KEY (NOMBRE, APELLIDOS)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

En al misma pestaña de estructura podemos ver los índices:

Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	NOMBRE APELLIDOS	0	A	No	

↓

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna
Editar Renombrar Eliminar	PRIMARY	BTREE	Sí	No	NOMBRE APELLIDOS

Tenemos un índice que está formado por dos campos, en la práctica que quiere decir pues si insertado datos y tengamos dos personas que se llaman igual el nombre y los apellidos no se van a poder agregar a la tabla.

No puede haber información repetida en los dos campos.

Este tipo de índice te obliga a no dejar vacíos los campos NOMBRE y APELLIDOS.

Ahora vamos con los índices ORDINARIOS, estos si que permiten duplicados pero si pueden estar vacíos. El comportamiento es como el de un campo normal pero nos va a permitir realizar búsquedas de una forma más rápida.

¿Cómo agregamos índices ORDINARIOS?

Primero vamos a borrar los índices.

```
1 ALTER TABLE EJEMPLO DROP PRIMARY KEY
```

localhost dice

¿Realmente desea ejecutar "ALTER TABLE EJEMPLO DROP PRIMARY KEY"?

Aceptar

Cancelar

Le damos a Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0004 segundos.)

Vamos a ver la estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Ya hemos eliminado las claves.

Imaginaros que reremos crear un índice ORDINARIO en el campo APELLIDOS para que las búsquedas, las consultas de este campo se realicen en un futuro con mayor rapidez.

La sintaxis sería la siguiente:

```
1 CREATE INDEX MIINDICE ON EJEMPLO (APELLIDOS)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS 	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Vamos a ver el índice.

Índices 

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna
 Editar  Renombrar  Eliminar	MIINDICE	BTREE	No	No	APELLIDOS

1 CREATE INDEX MIINDICE ON EJEMPLO (APELLIDOS)

Nombre que le dimos al índice MIINDICE y a que campo APELLIDOS.

A la hora de introducir la información como permite duplicados y permite dejarlo vacío es perfectamente posible tener dos personas con los mismos apellidos.

Siguiente índice ÚNICOS muy parecidos a los índices ordinarios pero no permiten duplicados, permite dejarlo vacío.

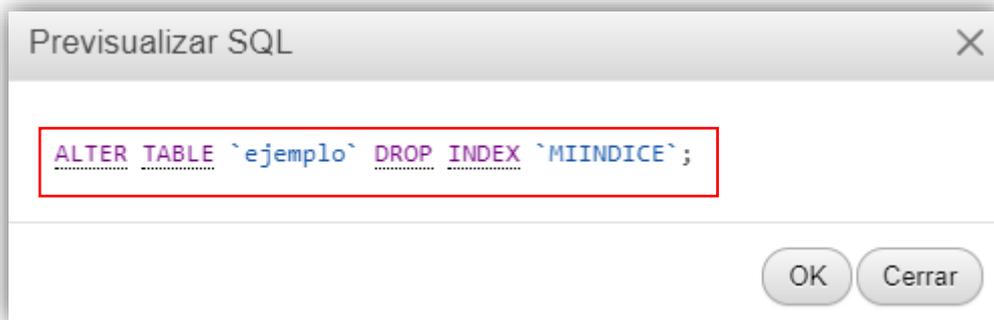
Vamos a eliminar el índice de nuevo.

En este caso dentro de la pestaña Estructura vamos a eliminar el índice.

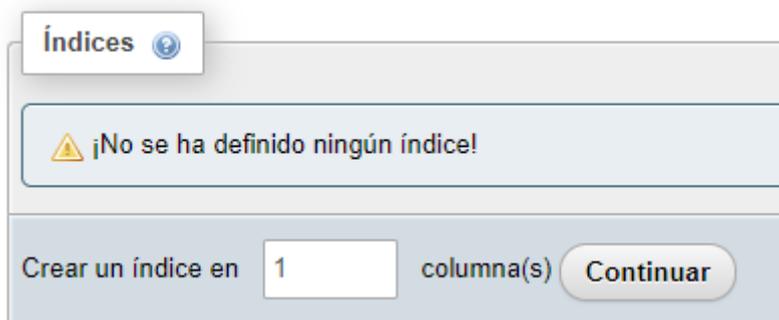
Índices 

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna
 Editar  Renombrar  Eliminar	MIINDICE	BTREE	No	No	APELLIDOS

Seleccionaremos Eliminar.



Nos dice como sería la instrucción SQL, le damos al botón OK.



Ya hemos eliminado el índice.

```
1 CREATE UNIQUE INDEX MIINDEX ON EJEMPLO(APELLIDOS)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

Vamos a ver la estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Vamos a ver el tipo de índice.

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
Editar Renombrar Eliminar	MIINDEX	BTREE	Sí	No	APELLIDOS	0	A	No	

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna
 Editar  Renombrar  Eliminar	MIINDEX	BTREE	Sí	No	APELLIDOS

Índice COMPUESTO formado por múltiples columnas y que permite dejar estos campos con la información vacía.

Ahora vamos a eliminar los índices del ejemplo anterior.

Índices 

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna
 Editar  Renombrar  Eliminar	MIINDEX	BTREE	Sí	No	APELLIDOS

Seleccionaremos Eliminar.



Contestaremos con OK.

Que sentido tiene crear índices COMPUESTOS en este ejemplo no podemos tener varias personas con los mismos NOMBRE y APELLIDOS. Pero si podemos dejar campos vacíos.

Vamos a crearlo.

```
1 CREATE UNIQUE INDEX MIINDEX ON EJEMPLO(NOMBRE, APELLIDOS)
```

Vamos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE 	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS 	varchar(20)	utf8mb4_general_ci		No	Ninguna

4	EDAD	tinyint(4)	Sí	NULL
---	------	------------	----	------

Vamos a ver los índices.

Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario	
 Editar	 Renombrar	 Eliminar	MIINDEX	BTREE	Sí	No	NOMBRE	0	A	No
							APELLIDOS	0	A	No

No permite duplicados.

Capítulo 19.- Eliminación de Índices

En el capítulo anterior ya eliminamos algunos registros pero en este lo vamos a profundizar.

En este capítulo veremos...

- Creación de índices
 - Eliminación índices

Vamos a trabajar con la tabla EJEMPLO.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Observamos como no tiene índices.

Vamos a crear un índice para posteriormente eliminarlo, tal y como vimos en el capítulo anterior.

Vamos a SQL.

```
1 CREATE INDEX MIINDICE ON EJEMPLO (DNI)
```

Vamos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI 	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Índices						
Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	
✎ Editar 📄 Renombrar 🗑 Eliminar	MIINDICE	BTREE	No	No	DNI	

Una vez hemos creado el índice para eliminarlo es tan sencillo como:

```
1 DROP INDEX MIINDICE ON EJEMPLO
```

No hace falta especificar el campo donde está el índice, seguimos en continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Este será el resultado en la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

Así como en el índice.

Índices

⚠ ¡No se ha definido ningún índice!

Crear un índice en columna(s) [Continuar](#)

El único que se diferencia en cuando es una clave primaria.

Vamos a crear una clave principal en el campo DNI para su posterior eliminación.

```
1 ALTER TABLE EJEMPLO ADD PRIMARY KEY(DNI)
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

Vamos a ver la estructura e índices de la tabla EJEMPLO.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	DNI 	varchar(15)	utf8mb4_general_ci		No	Ninguna
2	NOMBRE	varchar(20)	utf8mb4_general_ci		No	Ninguna
3	APELLIDOS	varchar(20)	utf8mb4_general_ci		No	Ninguna
4	EDAD	tinyint(4)			Sí	NULL

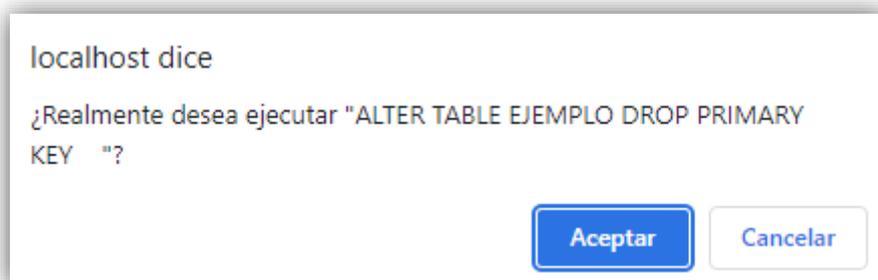
Los índices:

Índices						
Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	
 Editar  Renombrar  Eliminar	PRIMARY	BTREE	Sí	No	DNI	

Ahora vamos a eliminar la clave primaria.

```
1 ALTER TABLE EJEMPLO DROP PRIMARY KEY
```

Vamos a continuar.



Contestamos con Aceptar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Ya hemos eliminado la claven principal de la tabla EJEMPLO.

Capítulo 20.- Triggers I (Disparadores)

En este capítulo veremos...

- Qué son los triggers
 - Tipos
 - Utilidad
 - Creación

Imaginaros que tenemos una base de datos, y en esa base de datos tenemos una tabla. Nosotros queremos que al realiza una acción en concreto en esta tabla pues ocurra algo, se desencadene una acción en esta base de datos, esto es lo que se denomina un evento, un evento es una acción que se desencadena cuando ocurre algo y si hablamos de bases de datos pues un evento sería una acción que ocurre o se desencadena cuando ocurre algo en una tabla de tu base de datos.

Un Trigger es un objeto que se crea en tu base de datos y este objeto va a estar asociado a una tabla. Que quiere decir estar asociado a una tabla pues este Trigger desencadenará a una acción cuando ocurra algo en esta tabla en la que está asociado.

Que puede ocurrir en una tabla de una base de datos y que acción se puede desencadenar, pueden ocurrir básicamente tres cosas insertar registros (INSERT), actualizar registros (UPDATE) y también eliminar registros (DELETE). Un Trigger va a desencadenar una acción cuando ocurra alguna de estas tres cosas. Cuando alguien inserte información en esa tabla, cuando alguien actualice información en esa tabla o cuando alguien elimine información de esa tabla.

Entonces en definitiva ocurrirá una acción cuando alguien aplique la instrucción INSERT, UPDATE o DELETE.

¿Qué interés puede tener esto? , ¿Por qué voy a tener necesidad de que ocurra algo?, que se desencadene una acción en mi base de datos cuando un usuario inserta, actualiza o elimina datos de una tabla de la base de datos.

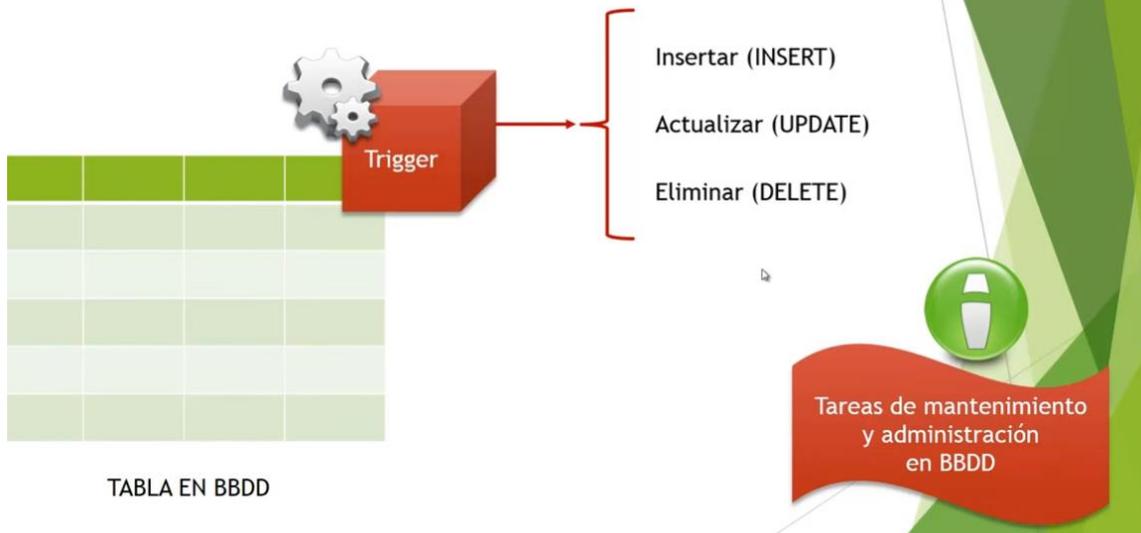
Imagínate que tu como administrador de esa base de datos necesitas llevar un registro de que usuario de los que trabajan en esta base de datos insertan registros y cuando los han insertado.

Pues esta tarea te la puede realizar un Trigger, lo que hará será desencadenar una acción después de que cada usuario haya insertado un registro en esa tabla y que acción va a desencadenar.

Por ejemplo copia o almacenar en otra tabla diferente el nombre del usuario que ha insertado el registro, el momento de insertarlo es decir la fecha día, mes y hora o cualquier otro tipo de información que se te pueda ocurrir y eso se hará de forma automática. Ese será el trabajo que se encargará el Trigger.

Otro ejemplo pues con la acción de actualizar que sabes que no tiene vuelta atrás en la tabla PRODUCTOS actualizamos los precios incrementando todos los precios un 5% ya no hay vuelta atrás es decir ya no puedo deshacer esta acción para ver cuales eran los precios anteriores a no ser que realice una consulta de acción para que realice una especie de respaldo, copia de seguridad, para que antes de realizar la actualización de la información esos registros se copien en otra tabla diferente y después de haberlos copiado en esta tabla, entonces si que procede realizar la actualización. De esa forma siempre vas a tener una copia de seguridad de cual era la información antes de actualizarse. Ya no digamos si vamos a borrar información de una base de datos.

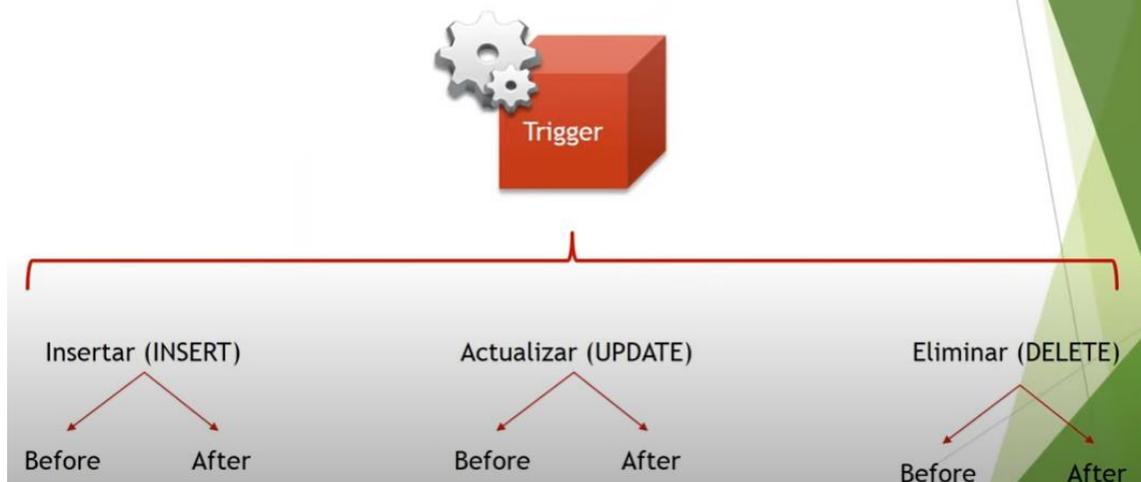
TRIGGERS



La pregunta es ¿Cuándo se va a desencadenar esta acción?.

TRIGGERS

Pero... ¿cuándo se ejecutan?



Nosotros podemos desencadenar la acción antes o después de insertar, antes o después de actualizar o antes o después de eliminar.

Supongamos que tenemos una tabla en la cual vamos a insertar registros y queremos almacenar en otra tabla diferente el nombre del usuario que ha insertado este registro, el campo clave del registro que ha insertado y la fecha y hora exacta en que ha insertado el registro. En este caso no tiene sentido antes si no tiene el registro insertado como voy a añadir el campo clave, el nombre de usuario y la fecha y hora en la tabla de respaldo si todavía no los ha insertado, en este caso se tendrá que realizar después.

Imaginaros otro caso diferente pues que a la hora de insertar personas que tienen un nombre, unos apellidos, una dirección, una edad, etc. y yo me quiero asegurar la edad de esas personas

que los usuarios están insertando en la tabla puedan ser negativas y quiero crear un Trigger que compruebe que la edad es siempre positiva de tal forma que no le deje insertar un registro cuya edad es negativa. En este ejemplo en concreto el Trigger tiene que ejecutarse antes o después, pues en este caso el Trigger tiene que ejecutarse antes.

Si esto lo trasladamos a actualizaciones o eliminación de datos, veremos que es muy importante elegir Before o After.

Por ejemplo imaginemos que tenemos dos tablas una clientes y otra de pedidos, esas dos tablas están relacionadas entre si con integridad referencial (no puede haber un pedido si el cliente no está dado de alta antes en la tabla clientes) si voy a dar de baja algún cliente antes tendré que comprobar si hay algún pedido relacionado, pues esta tarea de comprobación también se la podemos dejar a un Trigger y antes de eliminar el cliente de la tabla clientes tendré que mirar si hay algún pedido relacionado ya que no puede haber un pedido de un cliente que no existe.

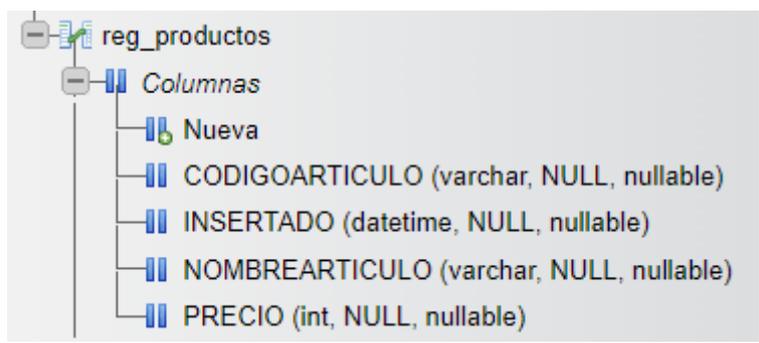
Vamos a crear un ejemplo muy sencillo, si os recordáis tenemos una tabla de PRODUCTOS, vamos a crear una segunda tabla vamos a guardar como un registro de que vamos insertando una vez creado esta segunda tabla vamos a crear un Trigger asociado a la tabla PRODUCTOS que después de haber insertado un registro rellene esta tabla nueva por ejemplo con el CODIGOARTICULO con el NOMBRE el PRECIO y en el momento que hemos insertado ese registro.

Vamos a crear la nueva tabla:

```
1 CREATE TABLE REG_PRODUCTOS (CODIGOARTICULO VARCHAR(25),
2                               NOMBREARTICULO VARCHAR(30),
3                               PRECIO INT(4), INSERTADO DATETIME)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	CODIGOARTICULO	varchar(25)	utf8mb4_general_ci		Sí	NULL
2	NOMBREARTICULO	varchar(30)	utf8mb4_general_ci		Sí	NULL
3	PRECIO	int(4)			Sí	NULL
4	INSERTADO	datetime			Sí	NULL

Ya hemos creado la tabla.

Vamos a crear un Trigger, que estará asociado a PRODUCTOS.

```
1 CREATE TRIGGER PRODUCTOS_AI AFTER INSERT ON PRODUCTOS FOR EACH ROW
2 INSERT INTO REG_PRODUCTOS(CODIGOARTICULO, NOMBREARTICULO, PRECIO, INSERTADO)
3 VALUES(NEW.CÓDIGOARTÍCULO, NEW.NOMBREARTÍCULO, NEW.PRECIO, NOW());
```

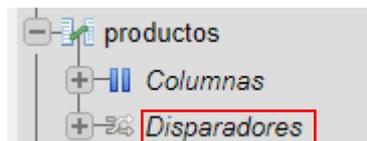
Para dar nombre a los Trigger se siguen unos criterios que no son obligatorios pero si recomendables empezamos por el nombre de la tabla que irá vinculada, seguid de un guion bajo, A Si es After o la letra B si es Before , seguido de la letra I para Inseter, U para update y D para delete.

A continuación el nombre de la tabla a la que estará vinculada, le decimos que se ejecute por cada registro añadido, no un conjunto de registros añadidos.

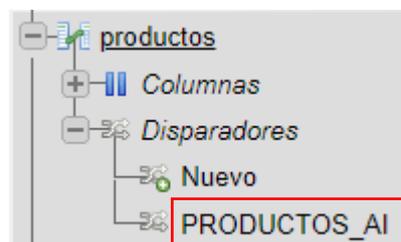
Que tiene que hacer cuando se ejecute insertar en la tabla REG_PRODUCTOS los siguientes campos CODIGOARTICULO, NOMBREARTICULO, PRECIO e INSERTADO con los valore nuevos introducidos en la tabla PRODUCTOS como NEW.CODIGOARTICULO, NEW.NOMBREARTICULO, NEW.PRECIO, y la fecha y hora actual NOW().

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0075 segundos.)



Si seleccionamos PRODUCTOS observaremos que ya tiene el Disparador, si lo seleccionamos.



Podemos crear un disparador nuevo o ver el que hemos creado, lo vamos a seleccionar.

Editar disparador

Detalles

Nombre del disparador: PRODUCTOS_AI

Tabla: productos

Tiempo: AFTER

Evento: INSERT

Definición:

```
INSERT INTO REG_PRODUCTOS(CODIGOARTICULO,
NOMBREARTICULO, PRECIO, INSERTADO)
VALUES(NEW.CÓDIGOARTÍCULO,
NEW.NOMBREARTÍCULO, NEW.PRECIO, NOW())
```

Definidor: root@localhost

[Continuar](#)

Aquí tenemos toda la información del disparador.

Ahora vamos a probar el disparador y es al agregar un registro en PRODUCTOS, la tabla REG_PRODUCTOS lo registrará.

```
1 INSERT INTO PRODUCTOS (CÓDIGOARTÍCULO, NOMBREARTÍCULO, PRECIO, PAÍSDEORIGEN)
2 VALUES('AR75', 'PANTALÓN', 50, 'ESPAÑA')
```

Le damos a continuar.

✓ 1 fila insertada. (La consulta tardó 0,0003 segundos.)

Ahora vamos a consultar la información que hay en la tabla REG_PRODUCTOS.

CODIGOARTICULO	NOMBREARTICULO	PRECIO	INSERTADO
AR75	PANTALÓN	50	2023-03-07 07:27:30

Además vamos a comprobar si se a agregado en PRODUCTOS:

CÓDIGOARTÍCULO	SECCIÓN	NOMBREARTÍCULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN	FOTO
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	0	ESPAÑA	NULL
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	0	ITALIA	NULL
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	0	MARRUECOS	NULL
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	0	USA	NULL
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	0	USA	NULL
AR07	CONFECCIÓN	SERRUCHO	30.2045	2001-03-23	0	FRANCIA	NULL
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	0	JAPÓN	NULL
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	0	MARRUECOS	NULL
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24	0	USA	NULL
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04	0	CHINA	NULL
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23	0	USA	NULL
AR13	CONFECCIÓN	CAMISA CABALLERO	67.1306	2002-08-11	0	ESPAÑA	NULL
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03	0	JAPÓN	NULL
AR15	CERÁMICA	PLATO DECORATIVO	54.0911	2000-06-07	0	CHINA	NULL
AR16	FERRETERÍA	ALICATES	6.7362	2000-04-17	0	ITALIA	NULL
AR17	JUGUETERÍA	MUÑECA ANDADORA	105.0593	2001-01-04	0	ESPAÑA	NULL
AR18	DEPORTES	PISTOLA OLÍMPICA	46.7347	2001-02-02	0	SUECIA	NULL
AR19	CONFECCIÓN	BLUSA SRA.	101.0566	2000-03-18	0	CHINA	NULL
AR20	CERÁMICA	JUEGO DE TE	43.2728	2001-01-15	0	CHINA	NULL
AR21	CERÁMICA	CENICERO	19.7468	2001-07-02	0	JAPÓN	NULL
AR22	FERRETERÍA	MARTILLO	11.3952	2001-09-04	0	ESPAÑA	NULL
AR23	CONFECCIÓN	CAZADORA PIEL	522.6930	2001-07-10	0	ITALIA	NULL
AR24	DEPORTES	BALÓN RUGBY	111.6440	2000-11-11	0	USA	NULL
AR25	DEPORTES	BALÓN BALONCESTO	75.2731	2001-06-25	0	JAPÓN	NULL
AR26	JUGUETERÍA	FUERTE DE SOLDADOS	143.7020	2000-11-25	0	JAPÓN	NULL
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05	0	ITALIA	NULL
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04	0	ESPAÑA	NULL
AR29	CONFECCIÓN	ABRIGO SRA	360.0736	2001-05-03	0	MARRUECOS	NULL
AR30	FERRETERÍA	DESTORNILLADOR	9.0584	2002-02-20	0	FRANCIA	NULL
AR31	JUGUETERÍA	PISTOLA CON SONIDOS	57.2500	2001-04-15	0	ESPAÑA	NULL
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03	0	USA	NULL
AR33	CERÁMICA	MACETA	29.0434	2000-02-23	0	ESPAÑA	NULL
AR34	OFICINA	PIE DE LÁMPARA	39.7606	2001-05-27	0	TURQUÍA	NULL
AR35	FERRETERÍA	LIMA GRANDE	22.0692	2002-08-10	0	ESPAÑA	NULL
AR36	FERRETERÍA	JUEGO DE BROCAS	15.0950	2002-07-04	0	TAIWÁN	NULL
AR37	CONFECCIÓN	CINTURÓN DE PIEL	4.3273	2002-05-12	0	ESPAÑA	NULL
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14	0	USA	NULL
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02	0	CHINA	NULL
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05	0	ESPAÑA	NULL
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02	0	ESPAÑA	NULL
AR75	NULL	PANTALÓN	50.0000	NULL	NULL	ESPAÑA	NULL

Capítulo 21.- Triggers II (Disparadores)

En este capítulo veremos...

- TRIGGER de actualización
- Diferencias entre AFTER Y BEFORE
- Uso de OLD y NEW

En que consiste un Trigger de actualización, supongamos que tenemos una tabla:

C Art	Nombre	Sección	Precio
Ar01	Pantalón	Confección	50
Ar02	Camisa	Confección	35
Ar03	Balón	Deporte	25
Ar04	Alicate	Ferretería	15
Ar05	Jarrón	Cerámica	90

TABLA EN BBDD

Nuestra tabla de PRODUCTOS con una serie de campos y registros, queremos realizar una sentencia SQL que va actualizar un registro o grupo de registros, queremos actualizar el artículos que tenemos seleccionado.

Imaginarios que la actualización va a consistir en incrementar el precio de este artículo, que ocurre cuando realizamos una sentencia SQL de estas características, como indicamos en el video anterior no hay vuelta atrás.

Siempre que ejecutas un INSERT, UPDATE o DELETE no hay vuelta atrás. Por este motivos los administradores o encargados de gestionar las bases de datos desean que se cree una especie de copia de respaldo antes de realizar cualquier acción de este tipo, en el caso que sea una actualización como se va a modificar la información que hay en las tablas, en este caso el precio muchas veces desean ver cual artículo ha sido modificado.

Aquí es donde entra el juego el Trigger.



El Trigger lo que haría antes de que se produzca la actualización guardar en otra tabla diferente.

TRIGGER de actualización



C Art	Nombre	Sección	Precio
Ar01	Pantalón	Confección	50
Ar02	Camisa	Confección	35
Ar03	Balón	Deporte	25
Ar04	Alicate	Ferretería	15
Ar05	Jarrón	Cerámica	90

TABLA EN BBDD

C Art	Nombre	Sección	Precio
Ar02	Camisa	Confección	45

TABLA RESPALDO

En este caso si vamos a incrementar el precio en 10 euros lo que haría Trigger sería guardar la información nueva del artículo que vamos a modificar. Así los responsables podrán ver el registro modificado.

Incluso con un Trigger de estas características podemos ir más allá, podemos incluir los campos que deseemos, por ejemplo podríamos incluir un campo más Precio anterior.

TRIGGER de actualización



C Art	Nombre	Sección	Precio
Ar01	Pantalón	Confección	50
Ar02	Camisa	Confección	45
Ar03	Balón	Deporte	25
Ar04	Alicate	Ferretería	15
Ar05	Jarrón	Cerámica	90

TABLA EN BBDD

C Art	Nombre	Sección	Precio	P. Ant
Ar02	Camisa	Confección	45	35

TABLA RESPALDO

Que refleje cual es el precio actual y el precio anterior antes de la actualización.

Y es aquí a la hora de realizar estas tareas donde entra en juego las sentencias OLD y NEW.

Si queremos realizar operaciones de este tipo tendremos que indicarle a esta tabla de respaldo en el campo "P. Ant" debe almacenare el valor OLD y en el campo "Precio" debe almacenarse el valor NEW.

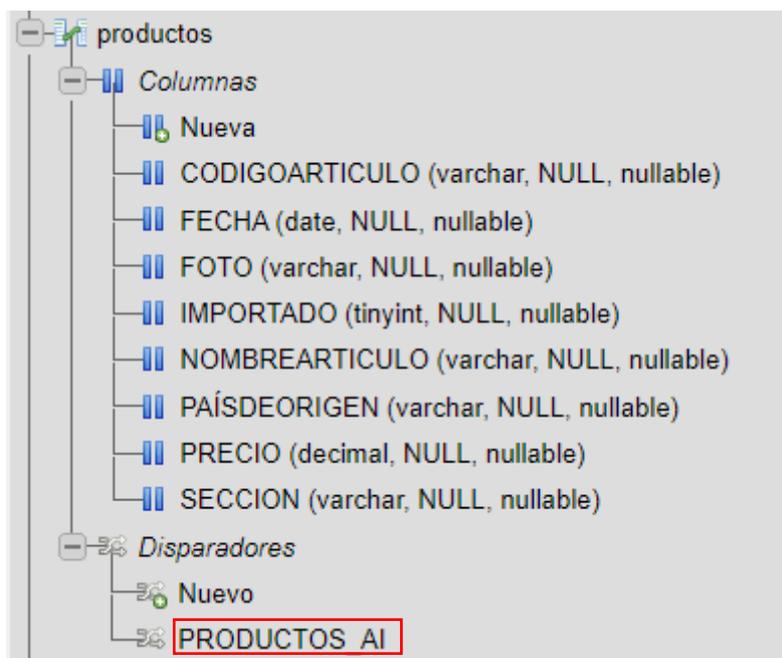
También podemos ver en este ejemplo la diferencia clara entre la sentencia AFTER y la sentencia BEFORE, este tipo de Trigger se puede realizar si usamos BEFORE.

Resumiendo si quieres realizar un respaldo del precio anterior tendremos que utilizar BEFORE.

También podemos agregar un campo que nos diga quién ha sido el usuario que ha realizado esta actualización, además de un campo de tipo fecha que refleje cuando se realizó esta actualización.

Así si se ha realizado una modificación indebidamente podremos saber que ha modificado, quien y cuando.

Vamos a verlo con un ejemplo:



Si os acordáis la tabla productos ya tiene un disparador que hicimos en el capítulo anterior, ahora antes de nada tenemos que crear la nueva tabla para controlar las actualizaciones.

```
1 CREATE TABLE PRODUCTOS_ACTUALIZADOS (ANT_CODIGOARTICULO VARCHAR(4), ANT_NOMBREARTICULO VARCHAR(25),
2 ANT_SECCIÓN VARCHAR(15), ANT_PRECIO INT(4), ANT_IMPORTADO VARCHAR(15),
3 ANT_PAISDEORIGEN VARCHAR(15), ANT_FECHA DATE,
4 NUE_CODIGOARTICULO VARCHAR(4), NUE_NOMBREARTICULO VARCHAR(25),
5 NUE_SECCION VARCHAR(15), NUE_PRECIO INT(4), NUE_IMPORTADO VARCHAR(15),
6 NUE_PAISDEORIGEN VARCHAR(15), NUE_FECHA DATE)
```

Creamos una tabla llamada PRODUCTOS_ACTUALIZADOS, que tendrá los siguientes campos:

Los campos antes de actualizar y los campos con sus actualizaciones, así sabremos en todo momento que actualización ha realizado.

Ahora vamos a modificar para agregar el usuario que ha realizado la actualización y fecha de cuando se hizo.

```

1 CREATE TABLE PRODUCTOS_ACTUALIZADOS (ANT_CODIGOARTICULO VARCHAR(4), ANT_NOMBREARTICULO VARCHAR(25),
2 ANT_SECCIÓN VARCHAR(15), ANT_PRECIO INT(4), ANT_IMPORTADO VARCHAR(15),
3 ANT_PAISDEORIGEN VARCHAR(15), ANT_FECHA DATE,
4 NUE_CODIGOARTICULO VARCHAR(4), NUE_NOMBREARTICULO VARCHAR(25),
5 NUE_SECCION VARCHAR(15), NUE_PRECIO INT(4), NUE_IMPORTADO VARCHAR(15),
6 NUE_PAISDEORIGEN VARCHAR(15), NUE_FECHA DATE,
7 USUARIO VARCHAR(15), F_MODIF DATE)

```

Estos campos cambiarlos a tipo decimal.

Vamos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Esta es la estructura de la nueva tabla:

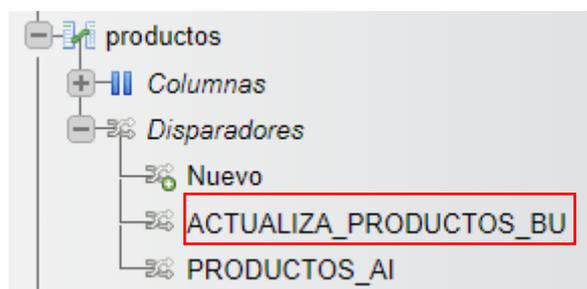
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	ANT_CODIGOARTICULO	varchar(4)	utf8mb4_general_ci		Sí	NULL
2	ANT_NOMBREARTICULO	varchar(25)	utf8mb4_general_ci		Sí	NULL
3	ANT_SECCIÓN	varchar(15)	utf8mb4_general_ci		Sí	NULL
4	ANT_PRECIO	int(4) ← Decimal			Sí	NULL
5	ANT_IMPORTADO	varchar(15)	utf8mb4_general_ci		Sí	NULL
6	ANT_PAISDEORIGEN	varchar(15)	utf8mb4_general_ci		Sí	NULL
7	ANT_FECHA	date			Sí	NULL
8	NUE_CODIGOARTICULO	varchar(4)	utf8mb4_general_ci		Sí	NULL
9	NUE_NOMBREARTICULO	varchar(25)	utf8mb4_general_ci		Sí	NULL
10	NUE_SECCION	varchar(15)	utf8mb4_general_ci		Sí	NULL
11	NUE_PRECIO	int(4) ← Decimal			Sí	NULL
12	NUE_IMPORTADO	varchar(15)	utf8mb4_general_ci		Sí	NULL
13	NUE_PAISDEORIGEN	varchar(15)	utf8mb4_general_ci		Sí	NULL
14	NUE_FECHA	date			Sí	NULL
15	USUARIO	varchar(15)	utf8mb4_general_ci		Sí	NULL
16	F_MODIF	date			Sí	NULL

Vamos a crear el Trigger.

```
1 CREATE TRIGGER ACTUALIZA_PRODUCTOS_BU BEFORE UPDATE ON PRODUCTOS FOR EACH ROW
2 INSERT INTO PRODUCTOS_ACTUALIZADOS
3 (ANT_CODIGOARTICULO, ANT_FECHA, ANT_IMPORTADO, ANT_NOMBREARTICULO,
4 ANT_PAISDEORIGEN, ANT_PRECIO, ANT_SECCIÓN,
5 NUE_CODIGOARTICULO, NUE_FECHA, NUE_IMPORTADO, NUE_NOMBREARTICULO,
6 NUE_PAISDEORIGEN, NUE_PRECIO, NUE_SECCION,
7 USUARIO, F_MODIF)
8 VALUES
9 (OLD.CODIGOARTICULO, OLD.FECHA, OLD.IMPORTADO, OLD.NOMBREARTICULO, OLD.PAISDEORIGEN, OLD.PRECIO, OLD.SECCION,
10 NEW.CODIGOARTICULO, NEW.FECHA, NEW.IMPORTADO, NEW.NOMBREARTICULO, NEW.PAISDEORIGEN, NEW.PRECIO, NEW.SECCION,
11 CURRENT_USER(), NOW())
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0052 segundos.)



Ya se ha creado el actualizador.

Este actualizador se ejecutará antes de actualizar.

Ahora nos queda realizar una consulta de actualización.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13
AR07	CONFECCIÓN	SERRUCHO	30.2045	2001-03-23
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10
AR10	JUGUETERÍA	CONSOLA VIDEO	442.5444	2002-09-24
AR11	CERÁMICA	TUBOS	168.4253	2000-02-04
AR12	FERRETERÍA	LLAVE INGLESA	24.3986	2001-05-23
AR13	CONFECCIÓN	CAMISA CABALLERO	67.1306	2002-08-11
AR14	JUGUETERÍA	TREN ELÉCTRICO	1505.3766	2001-07-03

Vamos a incrementar el precio del SERRUCHO en 20 euros.

```
1 UPDATE PRODUCTOS SET PRECIO=PRECIO+20 WHERE CODIGOARTICULO='AR07'
```

Le damos a continuar.

✓ 1 fila afectada. (La consulta tardó 0,0002 segundos.)

Vamos a ver la tabla Productos haber si se ha actualizado.

AR07	CONFECCIÓN SERRUCHO	50.2045	2001-03-23
------	---------------------	---------	------------

Ahora vamos a ver la tabla PRODUCTOS_ACTRUALIZADOS.

ANT_CODIGOARTICULO	ANT_NOMBREARTICULO	ANT_SECCIÓN	ANT_PRECIO	ANT_IMPORTADO	ANT_PAISDEORIGEN
AR07	SERRUCHO	CONFECCIÓN	30	0	FRANCIA

ANT_FECHA	NUE_CODIGOARTICULO	NUE_NOMBREARTICULO	NUE_SECCION	NUE_PRECIO	NUE_IMPORTADO
2001-03-23	AR07	SERRUCHO	CONFECCIÓN	50	0

NUE_PAISDEORIGEN	NUE_FECHA	USUARIO	F_MODIF
FRANCIA	2001-03-23	root@localhost	2023-03-07

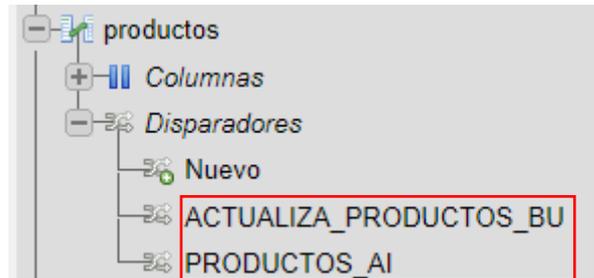
Como podemos observar la actualización a si del precio del SERRUCHO que antes de la actualización varía 30 después de la actualización vale 50, la actualización la ha realizado el usuario root@localhost el día 07-03-2023.

Capítulo 22.- Triggers III (Disparadores)

En este capítulo veremos...

- TRIGGER de eliminación.
- Eliminación de triggers.
- Modificación de triggers.

Vamos a ver cómo crear un Trigger de eliminación.



Vamos a seguir trabajando con la tabla de PRODUCTOS que en capítulos anteriores creamos dos triggers.

Un trigger de eliminación es especialmente útil que al ejecutarse cuando se elimina un registro de una tabla podemos realizar varias cosas entre ellas crear un copia de seguridad de este registro que se ha eliminado.

Para ello vamos que tener que crear una nueva tabla.

```
1 CREATE TABLE PROD_ELIMINADOS (C_ART VARCHAR(5), NOMBRE VARCHAR(15), SECCION VARCHAR(25),  
2 PRECIO DECIMAL(12,2), PAIS_ORIGEN VARCHAR(15))
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0006 segundos.)

Vamos a ver la estructura de la tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	C_ART	varchar(5)	utf8mb4_general_ci		Sí	NULL
2	NOMBRE	varchar(15)	utf8mb4_general_ci		Sí	NULL
3	SECCION	varchar(25)	utf8mb4_general_ci		Sí	NULL
4	PRECIO	decimal(12,2)			Sí	NULL
5	PAIS_ORIGEN	varchar(15)	utf8mb4_general_ci		Sí	NULL

El siguiente paso será crear el correspondiente TRIGGER.

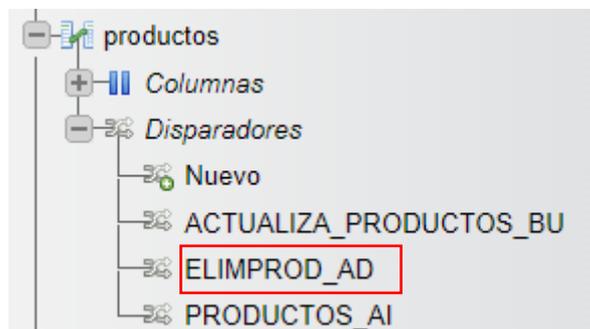
```

1 CREATE TRIGGER ELIMPROD_AD AFTER DELETE ON PRODUCTOS FOR EACH ROW
2 INSERT INTO PROD_ELIMINADOS (C_ART, NOMBRE, PAIS_ORIGEN, PRECIO, SECCION)
3 VALUE (OLD.CODIGOARTICULO, OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN, OLD.PRECIO, OLD.SECCION)

```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0044 segundos.)



Ya hemos creado el Trigger.

Ahora vamos a eliminar un registro a ver si funciona.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR27	CONFECCIÓN	ABRIGO CABALLERO	500000.0000	2002-04-05
AR28	DEPORTES	BALÓN FÚTBOL	43.9147	2002-07-04
AR29	CONFECCIÓN	ABRIGO SRA	360.0736	2001-05-03
AR30	FERRETERÍA	DESTORNILLADOR	9.0584	2002-02-20
AR31	JUGUETERÍA	PISTOLA CON SONIDOS	57.2500	2001-04-15
AR32	DEPORTES	CRONÓMETRO	439.1764	2002-01-03
AR33	CERÁMICA	MACETA	29.0434	2000-02-23
AR34	OFICINA	PIE DE LÁMPARA	39.7606	2001-05-27
AR35	FERRETERÍA	LIMA GRANDE	22.0692	2002-08-10
AR36	FERRETERÍA	JUEGO DE BROCAS	15.0950	2002-07-04
AR37	CONFECCIÓN	CINTURÓN DE PIEL	4.3273	2002-05-12
AR38	DEPORTES	CAÑA DE PESCA	270.0000	2000-02-14
AR39	CERÁMICA	JARRA CHINA	127.7704	2002-09-02
AR40	DEPORTES	BOTA ALPINISMO	144.0000	2002-05-05
AR41	DEPORTES	PALAS DE PING PONG	21.6000	2002-02-02
AR75	NULL	PANTALÓN	50.0000	NULL

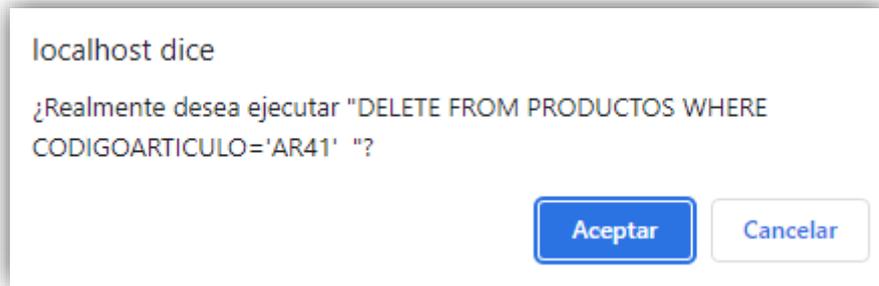
Queremos eliminar las palas de ping pong.

```

1 DELETE FROM PRODUCTOS WHERE CODIGOARTICULO='AR41'

```

Le damos a continuar.



Le damos Aceptar.

✓ 1 fila afectada. (La consulta tardó 0,0003 segundos.)

Si miramos la tabla PRODUCTOS observarás que este artículo ya no está.

Vamos a ver la tabla PROD_ELIMINADOS

C_ART	NOMBRE	SECCION	PRECIO	PAIS_ORIGEN
AR41	PALAS DE PING P	DEPORTES	21.60	ESPAÑA

Ahora nos damos cuenta de que en esta tabla falta el usuario que eliminó este registro y la fecha de cuando lo hizo.

Está claro que hay que modifica dos cosas, la tabla PROD_ELIMINADOS, agregando los campos que faltan y a continuación el Trigger para que almacene esta información en la tabla.

Vamos a agregar dos campos a la tabla PROD_ELIMINADOS.

```
1 ALTER TABLE PROD_ELIMINADOS ADD COLUMN (USUARIO VARCHAR(15), FECHA_MODIF DATE)
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver la estructura de la tabla PROD_ELIMINADOS.

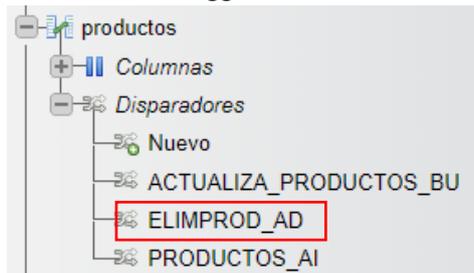
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	C_ART	varchar(5)	utf8mb4_general_ci		Sí	NULL
2	NOMBRE	varchar(15)	utf8mb4_general_ci		Sí	NULL
3	SECCION	varchar(25)	utf8mb4_general_ci		Sí	NULL
4	PRECIO	decimal(12,2)			Sí	NULL

5	PAIS_ORIGEN	varchar(15)	utf8mb4_general_ci	Sí	NULL
6	USUARIO	varchar(15)	utf8mb4_general_ci	Sí	NULL
7	FECHA_MODIF	date		Sí	NULL

Ya hemos agregado los campos.

Ahora tenemos que modificar el Trigger para decirle que hemos agregado dos campos en la tabla PROD_ELIMINADOS.

Primero vamos a ver como se modifica el Trigger desde la interface gráfica.



Seleccionamos el Trigger que queremos modificar.

Editar disparador

Detalles

Nombre del disparador:

Tabla:

Tiempo:

Evento:

Definición:

```

INSERT INTO PROD_ELIMINADOS (C_ART, NOMBRE,
PAIS_ORIGEN, PRECIO, SECCION)
VALUE (OLD.CODIGOARTICULO,
OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN,
OLD.PRECIO, OLD.SECCION)

```

Definidor:

Realizaremos las siguientes modificaciones:

Editar disparador

Detalles

Nombre del disparador:

Tabla:

Tiempo:

Evento:

Definición:

```
INSERT INTO PROD_ELIMINADOS (C_ART, NOMBRE, PAIS_ORIGEN, PRECIO, SECCION, USUARIO, FECHA_MODIF)
VALUE (OLD.CODIGOARTICULO, OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN, OLD.PRECIO, OLD.SECCION, CURRENT_USER(), NOW())
```

Definidor:

Seguido del botón Continuar.

✓ Se modificó el disparador `ELIMPROD_AD`.

✓ Se modificó el disparador `ELIMPROD_AD`.

Y nos muestra la nueva sintaxis:

```
DROP TRIGGER IF EXISTS `ELIMPROD_AD`;CREATE DEFINER='root'@'localhost' TRIGGER `ELIMPROD_AD` AFTER DELETE ON `productos` FOR EACH ROW INSERT INTO PROD_ELIMINADOS (C_ART, NOMBRE, PAIS_ORIGEN, PRECIO, SECCION, USUARIO, FECHA_MODIF) VALUE (OLD.CODIGOARTICULO, OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN, OLD.PRECIO, OLD.SECCION, CURREENT_USER(), NOW())
```

```
DROP TRIGGER IF EXISTS `ELIMPROD_AD`;CREATE DEFINER='root'@'localhost' TRIGGER `ELIMPROD_AD` AFTER DELETE ON `productos` FOR EACH ROW INSERT INTO PROD_ELIMINADOS (C_ART, NOMBRE, PAIS_ORIGEN, PRECIO, SECCION, USUARIO, FECHA_MODIF)
```

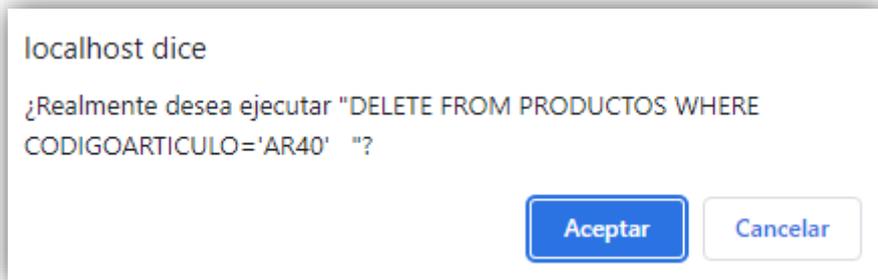
```
VALUE (OLD.CODIGOARTICULO, OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN, OLD.PRECIO, OLD.SECCION, CURREENT_USER(), NOW())
```

Ahora vamos a eliminar un segundo producto para ver si además guarda el usuario y fecha de la eliminación.

Vamos a eliminar el artículo con el código AR40 que son unas BOTAS DE ALPINISMO.

```
1 DELETE FROM PRODUCTOS WHERE CODIGOARTICULO='AR40'
```

Le damos a continuar.



Le damos a Aceptar.

✓ 1 fila afectada. (La consulta tardó 0,0003 segundos.)

Ahora vamos a consultar la tabla PROD_ELIMINADOS.

C_ART	NOMBRE	SECCION	PRECIO	PAIS_ORIGEN	USUARIO	FECHA_MODIF
AR41	PALAS DE PING P	DEPORTES	21.60	ESPAÑA	NULL	NULL
AR40	BOTA ALPINISMO	DEPORTES	144.00	ESPAÑA	root@localhost	2023-03-07

Ahora si guarda el USUARIO y FECHA_MODIF.

Como haremos la modificación desde código SQL.

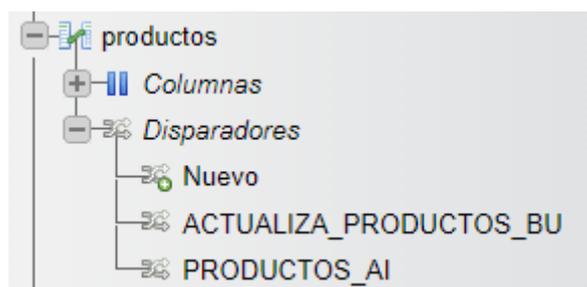
Primero eliminaríamos el Trigger.

```
1 DROP TRIGGER ELIMPROD_AD
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0023 segundos.)

Le damos a actualizar.



Ya hemos eliminado el Trigger.

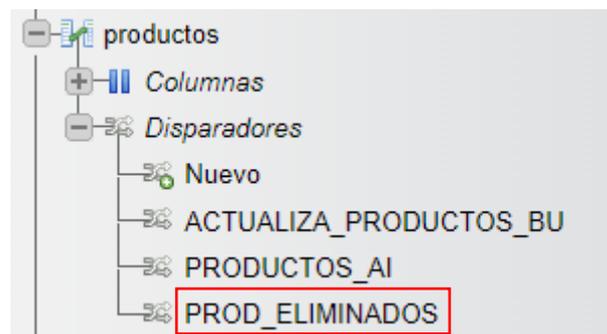
Vamos a volverlo a crear.

```
1 CREATE TRIGGER PROD_ELIMINADOS AFTER DELETE ON PRODUCTOS FOR EACH ROW
2 INSERT INTO PROD_ELIMIANADOS (C_ART, NJOMBRE, PAIS_ORIGEN, PRECIO, SECCION, USUARIO, FECHA_MODIF)
3 VALUE(OLD.CODIGOARTICULO, OLD.NOMBREARTICULO, OLD.PAÍSDEORIGEN, OLD.PRECIO, OLD.SECCION,
4       CURRENT_USER(), NOW())
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0042 segundos.)

Vamos a ver si se ha agregado a la tabla PRODUCTOS.



Otro ejemplo de cómo hacerlo.

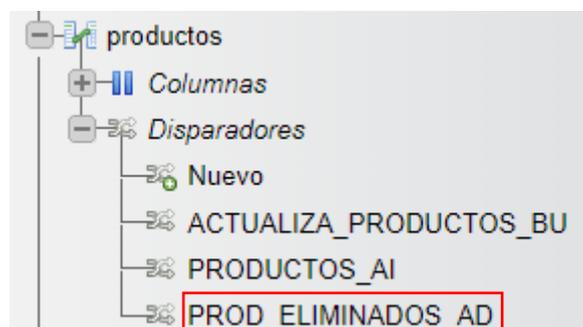
```
1 DROP TRIGGER IF EXISTS PROD_ELIMINADOS;
2 CREATE TRIGGER PROD_ELIMINADOS_AD AFTER DELETE ON PRODUCTOS FOR EACH ROW
3 INSERT INTO PROD_ELIMINADOS (C_ART, FECHA_MODIF, NOMBRE, SECCION)
4 VALUES(OLD.CODIGOARTICULO, NOW(), OLD.NOMBREARTICULO, OLD.SECCION)
```

Le damos a Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0032 segundos.)

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0052 segundos.)

Vamos a ver la tabla:



Ya hemos modificado el Trigger y hemos cambiado el nombre del mismo.

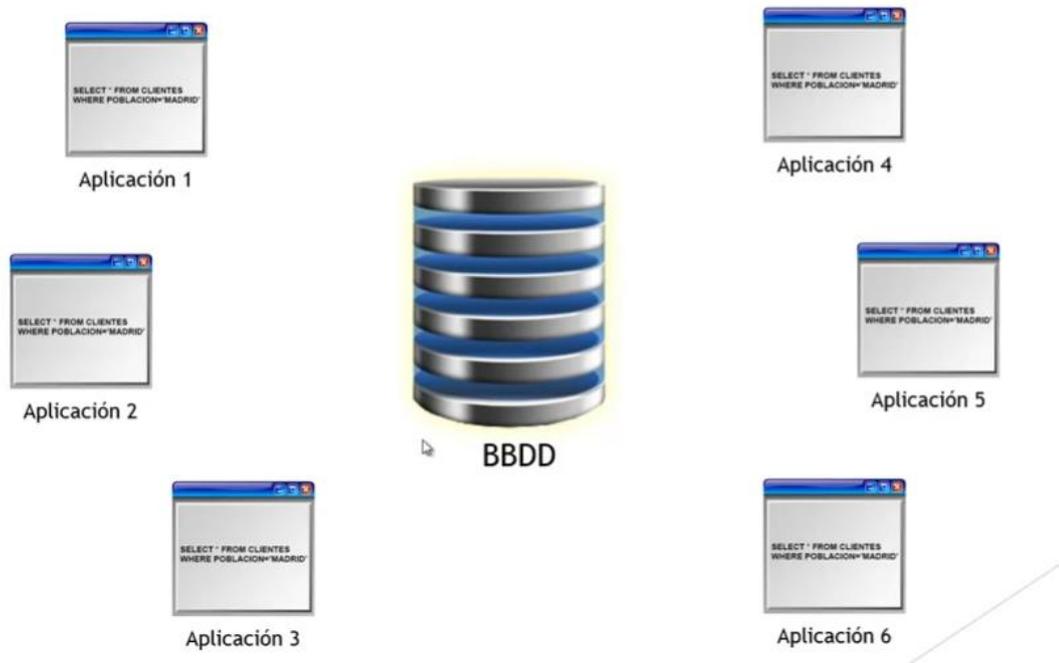
Capítulo 23.- Procedimientos almacenados I

En este capítulo veremos...

- ¿Qué es un procedimiento almacenado? Utilidad de un PA
- Ejemplos de PA sencillos



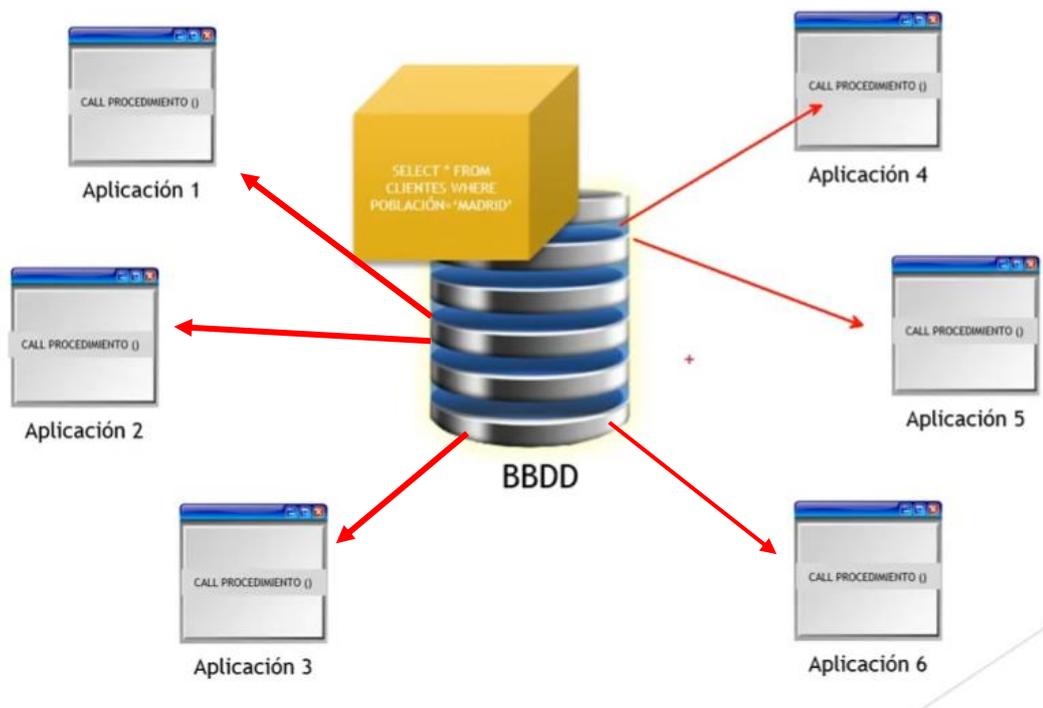
Vamos a imaginarnos que trabajamos contra una base de datos, nos imaginamos que nos encontramos en un entorno de trabajo, oficina, centro de estudios, etc., en el cual hay varias personas, con varias aplicaciones trabajando.



Supongamos que hay 6 usuarios utilizando aplicaciones diferentes sobre la misma base de datos. Además están haciendo consultas sobre la misma tabla.

Cuando nos encontramos en esta situación es muy útil trabajar con procesamientos almacenados, por cuestiones de eficiencia y por cuestiones de seguridad.

En que consiste un procedimiento almacenado y como puede ayudar a nuestra eficiencia y a nuestra seguridad, un procedimiento almacenado consiste almacenar el procedimiento estas instrucciones repetitivas en SQL que se ejecutan desde diferentes aplicaciones y diferentes puestos, diferentes usuarios.



Creamos en el servidor y almacenamos en el servidor un procedimiento con la instrucción y después lo único que tendremos que hacer es en vez de ejecutar en cada aplicación esa instrucción SQL, únicamente llamar a este procedimiento desde todas las aplicaciones.

Para poder hacer esto se necesitan ciertos conocimientos de programación, saber lo que es una variable, saber lo que es un parámetro, lo que es un paso de valor por parámetro, también lo que es un bucle, lo que es un condicional y vendrían muy bien si estos conocimientos de programación en Visual Basic o en Pascal.

Vamos a realizar un procedimiento para consultar a los clientes que pertenecen a una determinada población.

Como creamos en nuestra base de datos un procedimiento almacenado al cual podemos llamar para ejecutar su instrucción.

Vamos a SQL.

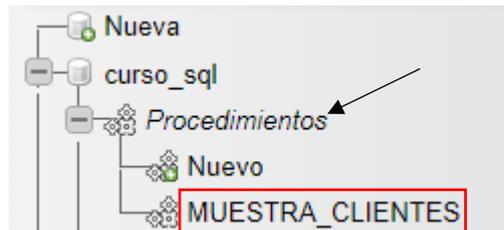
```
1 CREATE PROCEDURE MUESTRA_CLIENTES()  
2 SELECT * FROM CLIENTES WHERE POBLACIÓN='MADRID'
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0029 segundos.)



Le damos a actualizar.



Este es el procedimiento que hemos creado.

Si seleccionamos donde pone Procedimientos.

	Nombre	Tipo	Retorna	
<input type="checkbox"/>	MUESTRA_CLIENTES	PROCEDURE		Editar Ejecutar Exportar Eliminar

Desde aquí lo podemos editar, ejecutar, exportar o eliminar, vamos a darle a ejecutar.

CÓDIGO CLIENTE	EMPRESA	DIRECCIÓN	POBLACIÓN	TELÉFONO	RESPONSABLE
CT01	BELTRÁN E HIJOS	LAS FUENTES 78	MADRID	(914) 45- 64- 35	ANGEL MARTÍNEZ
CT04	EXPORTASA	VALLECAS 34	MADRID	(913) 45- 23- 78	ELVIRA GÓMEZ
CT07	LA CASA DEL JUGUETE	AMÉRICA 45	MADRID	(912) 64- 99- 87	ELÍAS PÉREZ
CT10	FERRETERÍA EL CLAVO	PASEO DE ÁLAMOS 78	MADRID	(914) 35- 48- 66	MANUEL MENÉNDEZ
CT14	DEPORTES GARCÍA	GUZMÁN EL BUENO 45	MADRID	(913) 29- 94- 75	ANA JIMÉNEZ
CT19	CONFECCIONES GALÁN	FUENCARRAL 78	MADRID	(913) 85- 92- 34	JUAN GARCÍA
CT23	EL PALACIO DE LA MODA	ORTEGA Y GASSET 129	MADRID	(927) 78- 52- 35	LAURA CARRASCO
CT26	FERRETERÍA LA ESCOBA	ORENSE 7	MADRID	(918) 45- 93- 46	JOSÉ GARCÍA
CT30	BAZAR EL ARGENTINO	ATOCHA 55	MADRID	(912) 49- 59- 73	ADRIÁN ÁLVAREZ
CT34	BAZAR LA FARAONA	CASTILLA Y LEÓN 34	MADRID	(915) 48- 36- 27	ANGEL SANTAMARÍA
CT36	JUGUETES EDUCATIVOS SANZ	ORENSE 89	MADRID	(916) 87- 23- 54	PEDRO IGLESIAS
CT39	FERRETERÍA LIMA	VALLECAS 45	MADRID	(913) 53- 27- 85	LUIS GARCÍA

Nos muestra a los clientes que son de Madrid.

En la parte superior nos indica como se llama a un procedimiento desde SQL.

```
CALL `MUESTRA_CLIENTES` ();
```

```
1 CALL MUESTRA_CLIENTES()
```

Le damos a continuar.

El resultado será el mismo.

Vamos a realizar un segundo ejemplo, en este caso va a recibir parámetros.

En este caso queremos que actualice el precio de algún artículo en concreto.

```
1 CREATE PROCEDURE ACTUALIZA_PRODUCTOS(N_PRECIO DECIMAL(12,3), CODIGO VARCHAR(4))
2 UPDATE PRODUCTOS SET PRECIO=N_PRECIO WHERE CODIGOARTICULO=CODIGO;
```

Le damos a Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0041 segundos.)

Vamos a consultar la tabla PRODUCTOS.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA	IMPORTADO	PAÍSDEORIGEN
AR01	FERRETERÍA	DESTORNILLADOR	6.6280	2000-10-22	0	ESPAÑA
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11	0	ITALIA
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26	0	MARRUECOS
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20	0	USA
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13	0	USA
AR07	CONFECCIÓN	SERRUCHO	50.2045	2001-03-23	0	FRANCIA
AR08	JUGUETERÍA	CORREPASILLOS	103.3356	2000-04-11	0	JAPÓN
AR09	CONFECCIÓN	PANTALÓN SEÑORA	174.2310	2000-01-10	0	MARRUECOS

Queremos que el artículo con el código 'AR01' su nuevo precio sea 10.15.

Vamos a SQL.

```
1 CALL ACTUALIZA_PRODUCTOS(10.15, 'AR01')
```

Le damos a continuar.

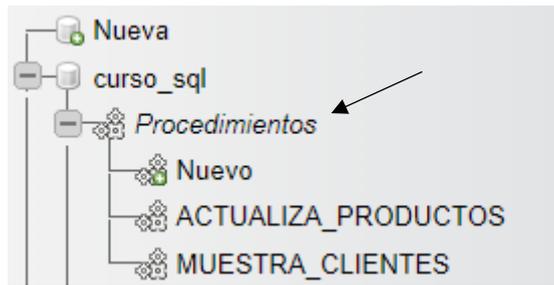
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0042 segundos.)

Vamos a consultar el registro AR01 para comprobar si ha cambiado el precio.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	10.1500	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13

Podrás comprobar que de 6.6280 a pasado a valer 10.1500.

También lo podemos ejecutar desde el panel de procedimientos.



Seleccionamos Procedimientos.

	Nombre	Tipo	Retorna			
<input type="checkbox"/>	ACTUALIZA_PRODUCTOS	PROCEDURE		Editar	Ejecutar	Exportar Eliminar
<input type="checkbox"/>	MUESTRA_CLIENTES	PROCEDURE		Editar	Ejecutar	Exportar Eliminar

Muestra los dos procedimientos, vamos a ejecutar ACTUALIZAR_PRODUCTOS.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	10.1500	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26
AR04	DEPORTES	RAQUETA TENIS	93.4694	2000-03-20
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13

Queremos que la RAQUETA TENIS tenga un precio de 105,25.

Ejecutar rutina `ACTUALIZA_PRODUCTOS`

Parámetros de rutina

Nombre	Tipo	Función	Valor
N_PRECIO	DECIMAL	<input type="text"/>	105.25
CODIGO	VARCHAR	<input type="text"/>	AR04

Ponemos el nuevo valor que tendrá el producto y el código que queremos cambiar, seguido del botón continuar.

Vamos a consultar los productos:

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	10.1500	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26
AR04	DEPORTES	RAQUETA TENIS	105.2500	2000-03-20
AR06	DEPORTES	MANCUERNAS	60.0000	2000-09-13

Así se trabaja más eficiente.

Capítulo 24.- Procedimientos y triggers

En este capítulo veremos...

- Declaración de variables en procedimientos almacenados.
- Triggers condicionales.

Vamos a crear un procedimiento nuevo que va a tener poca utilidad práctica, la única utilidad que va a tener es para que se entiendan bien algunas estructuras importantes que podemos crear dentro de un procedimiento, como por ejemplo la declaración de variables o lo que es bloque de ejecuciones.

Nos vamos a SQL y vamos a crear un procedimiento de almacenamiento capaz de averiguar la edad en función del año de nacimiento que nosotros le pasemos por parámetro.

```
1 DELIMITER $$
2 CREATE PROCEDURE CALCULA_EDAD(AGNO_NACIMIENTO INT)
3     BEGIN
4         DECLARE AGNO_ACTUAL INT DEFAULT 2023;
5         DECLARE EDAD INT;
6         SET EDAD= AGNO_ACTUAL-AGNO_NACIMIENTO;
7         SELECT EDAD;
8     END;$$
9 DELIMITER ;
```

En la línea 1 declaramos el delimitador de bloque que en este caso son \$\$

En la línea 2 creamos un procedimiento llamado CALCULA_EDAD con el argumento AGNO_NACIMIENTO de tipo entero.

En la línea 3 con BEGIN le decimos donde empieza el bloque.

En la línea 4 declaramos la variable AGNO_ACTUAL de tipo entero con el valor de 2023.

En la línea 5 declaramos la variable EDAD de tipo entero.

En la línea 6 a la variable EDAD le asignamos el valor sobre el cálculo de AGNO_ACTUAL – AGNO_NACIMIENTO.

En la línea 7 retornamos el valor de EDAD, se utiliza SELECT como RETURN.

En la línea 8 con END delimitamos el bloque por la parte final.

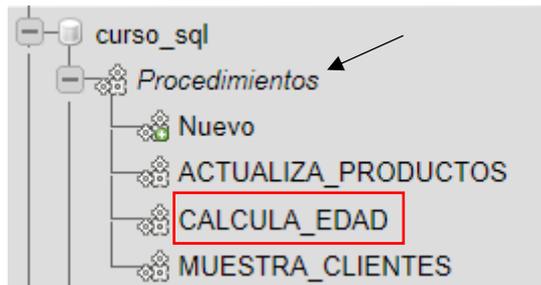
En la línea 9 reseteamos el delimitador.

En cada instrucción utilizamos el delimitador por defecto que es el punto y coma ‘;’.

Seleccionamos el botón Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0035 segundos.)

Se ha realizado correctamente.



Seleccionamos procedimientos.

	Nombre	Tipo	Retorna	
<input type="checkbox"/>	ACTUALIZA_PRODUCTOS	PROCEDURE		<input type="text" value="Editar"/> <input type="text" value="Ejecutar"/> <input type="text" value="Exportar"/> <input type="text" value="Eliminar"/>
<input type="checkbox"/>	CALCULA_EDAD	PROCEDURE		<input type="text" value="Editar"/> <input type="text" value="Ejecutar"/> <input type="text" value="Exportar"/> <input type="text" value="Eliminar"/>
<input type="checkbox"/>	MUESTRA_CLIENTES	PROCEDURE		<input type="text" value="Editar"/> <input type="text" value="Ejecutar"/> <input type="text" value="Exportar"/> <input type="text" value="Eliminar"/>

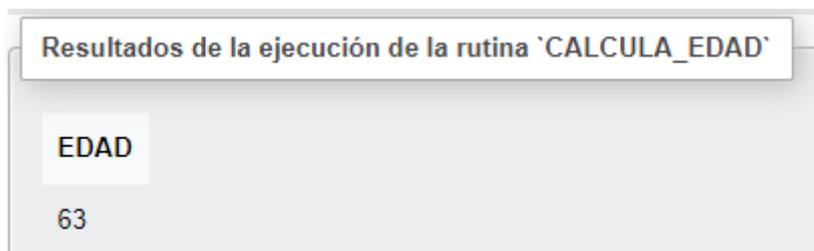
Y lo ejecutamos.

Ejecutar rutina `CALCULA_EDAD`

Parámetros de rutina

Nombre	Tipo	Función	Valor
AGNO_NACIMIENTO	INT	<input type="text" value=""/>	<input type="text" value="1960"/>

Contestamos por mi año de nacimiento 1960, seguido de Continuar.



Por ejemplo imaginemos los siguiente, queremos crear un Trigger para nuestra tabla de PRODUCTOS la cual ya tiene declarado otros Triggers.

Imaginemos que a la hora de trabajar con esta tabla es frecuente que hagamos actualizaciones de precios, pero de alguna forma queremos establecer un control en estos precios, no queremos que se realicen actualizaciones con precios disparados.

Con Update quiero controlar que el precio no sea negativo y tampoco que no tenga un precio disparado.

La condición es que el precio tiene que estar entre 0 y 1000.

Si el precio es negativo que lo deje en 0 y si el precio es mayor a 1000 que lo deje a 1000.

Vamos a SQL.

```
1 DELIMITER $$
2 CREATE TRIGGER REvisa_PRECIO_BU BEFORE UPDATE ON PRODUCTOS FOR EACH ROW
3 BEGIN
4     IF (NEW.PRECIO < 0) THEN
5         SET NEW.PRECIO = 0;
6     ELSEIF (NEW.PRECIO > 1000) THEN
7         SET NEW.PRECIO = 1000;
8     END IF;
9
10
11 END; $$
12 DELIMITER ;
```

En la línea 1 definimos cual será el delimitador de bloques en este caso es \$\$

En la línea 2 creamos un Trigger con el nombre REvisa_PRECIO_BU queremos que se ejecute antes de actualizar en la tabla PRODUCTOS.

En la línea 3 indicamos inicio de bloque con BEGIN.

En la línea 4 con un condicional comparamos si NEW.PRECIO es menor de 0.

En la línea 5 si la condición de la línea anterior se cumple que valor de NEW.PRECIO será igual a 0.

En la línea 6 con el condicional comprobamos que NEW.PRECIO es mayor de 1000.

En la línea 7 si la condición de la línea anterior se cumple el valor de NEW.PRECIO será igual a 1000.

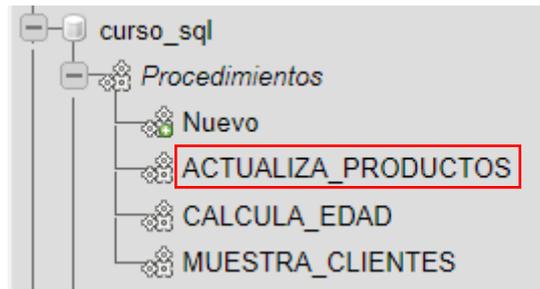
En la línea 8 fin del bloque condicional.

En la línea 11 Con END delimitador de bloque.

En la línea 12 reseteamos el delimitador.

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0045 segundos.)



El artículo AR01 DESTORNILLADOR vemos que vale 6.63 lo vamos a actualizar a 15 euros.

```
1 UPDATE PRODUCTOS SET PRECIO=15 WHERE CODIGOARTICULO='AR01'
```

Le damos a continuar.

✓ 1 fila afectada. (La consulta tardó 0,0004 segundos.)

Este será el nuevo precio del DESTONILLADOR.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	15.0000	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26

Vamos a realizar una consulta con un valor superior a 1000.

```
1 UPDATE PRODUCTOS SET PRECIO=8500 WHERE CODIGOARTICULO='AR01'
```

Le damos a continuar.

✓ 1 fila afectada. (La consulta tardó 0,0014 segundos.)

Vamos a ver el valor del DESTORNILLADOR.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	1000.0000	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26

Como nos hemos excedido en el precio el a puesto el precio máximo que es de 1000 euros.

Otra opción sería que cuando pongas un precio mayor de 1000 euros lo deje con el precio actual.

Vamos a poner el DESTONILLADOR A 6,83 euros.

```
1 UPDATE PRODUCTOS SET PRECIO=6.83 WHERE CODIGOARTICULO='AR01'
```

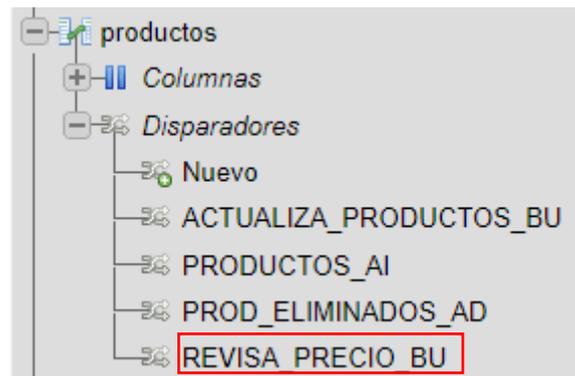
Le damos a continuar.

✓ 1 fila afectada. (La consulta tardó 0,0002 segundos.)

Vamos a ver el precio del DESTORNILLADOR.

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	6.8300	2000-10-22
AR02	CONFECCIÓN	TRAJE CABALLERO	284.5769	2002-03-11
AR03	JUGUETERÍA	COCHE TELEDIRIGIDO	159.4462	2002-05-26

Seleccionaremos el Trigger que tenemos que modificar.



Seleccionamos REVISAR_PRECIO_BU.

Editar disparador

Detalles

Nombre del disparador:

Tabla:

Tiempo:

Evento:

Definición:

```
BEGIN
  IF(NEW.PRECIO<0) THEN
    SET NEW.PRECIO=OLD.PRECIO;
  ELSEIF(NEW.PRECIO>1000) THEN
    SET NEW.PRECIO=OLD.PRECIO;
  END IF;
END
```

Definidor:

Si ponemos un precio inferior a 0 o superior a 1000 dejar el precio actual.
Le damos a continuar.

✓ Se modificó el disparador `REVISA_PRECIO_BU`.

✓ Se modificó el disparador `REVISA_PRECIO_BU`.

Ahora volvemos a ejecutar una consulta de tipo UPDATE.

```
1 UPDATE PRODUCTOS SET PRECIO=8500 WHERE CODIGOARTICULO='AR01'
```

Le damos a continuar.

✓ 0 filas afectadas. (La consulta tardó 0,0003 segundos.)

CODIGOARTICULO	SECCION	NOMBREARTICULO	PRECIO	FECHA
AR01	FERRETERÍA	DESTORNILLADOR	6.8300	2000-10-22

Lo ha dejado al mismo precio.

Capítulo 25.- Vistas (Final curso)

En este capítulo veremos...

- ¿Qué son las vistas?
- Creación de vistas

VISTAS (VIEWS)

▶ VENTAJAS:

- ▶ Privacidad de la información
- ▶ Optimización de la BBDD
- ▶ Entorno de pruebas



BBDD

La primer ventaja es la privacidad de la información, ya que puedes crear diferentes perfiles de usuario en la base de datos de tal forma que puedes ocultar lo que son las tablas a un grupo de usuario y que solamente tengan acceso a lo que son las vistas.

Otra ventaja es la optimización de la base de datos, si tienes a un grupo de usuarios que están todo el día creando el mismo tipo de consulta sobre las mismas tablas para que la base de datos esté mejor optimizada y el rendimiento sea mejor lo que haces es crear una vista de tal forma que se consulte esta vista antes de genera consultas constantes y complejas de las tablas de la base de datos.

Otra ventaja es que a los desarrolladores les ofrece un entorno de pruebas en el que puedan comprobar que datos existen incluso hacer algunas pruebas sin que corra peligro la integridad de los datos.

Como se crea una vista con el gestor de datos MySQL.

Las vistas no son exclusivas de este gestor de base de datos.

Pongámonos en la siguiente situación, tengo un grupo de usuarios haciendo constantemente consultas en la tabla de PRODUCTOS y que estas consultas o instrucciones SQL hace referencia constantemente a la SECCION de los artículos.

Lo que podemos es crear una vista que es como un acceso directo o consulta de Access, algo que está vinculado con tabla origen pero que no obliga al motor de la base de datos a trabajar cada vez que se pide esta información.

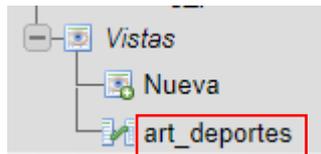
Vamos a crear una primera vista que nos ofrezca tres columnas que son las que necesitamos ver en cada momento y además que nos muestre los artículos de la sección de deportes.

```
1 CREATE VIEW ART_DEPORTES AS
2 SELECT NOMBREARTICULO, SECCION, PRECIO FROM PRODUCTOS
3 WHERE SECCION='DEPORTES'
```

Seleccionaremos continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

En el panel de la izquierda aparece un nuevo objeto que son las vistas.



Que ventaja tiene esto, pues cada vez que quieras consultar esta base de datos los artículos de DEPORTE no tienes que hacer una instrucción SQL nueva y hacer trabajar al motor de la base de datos.

Selecciona la vista.

NOMBREARTICULO	SECCION	PRECIO
RAQUETA TENIS	DEPORTES	105.2500
MANCUERNAS	DEPORTES	60.0000
PISTOLA OLÍMPICA	DEPORTES	46.7347
BALÓN RUGBY	DEPORTES	111.6440
BALÓN BALONCESTO	DEPORTES	75.2731
BALÓN FÚTBOL	DEPORTES	43.9147
CRONÓMETRO	DEPORTES	439.1764
CAÑA DE PESCA	DEPORTES	270.0000

Y ya nos muestra los datos.

Estas vistas son actualizables, si yo hago alguna actualización en la tabla de PRODUCTOS, esta se reflejará en la vista.

Por ejemplo tenemos una RAQUETA TENIS que cuesta 105,25 euros.

Vamos a realizar una modificación que consiste en incrementar 10 euros la RAQUETA TENIS.

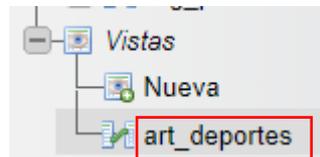
Vamos a SQL.

```
1 UPDATE PRODUCTOS SET PRECIO = PRECIO+10
2 WHERE NOMBREARTICULO='RAQUETA TENIS'
```

Vamos a continuar.

✓ 1 fila afectada. (La consulta tardó 0,0018 segundos.)

Ahora vamos a seleccionar la vista que hemos creado.



NOMBREARTICULO	SECCION	PRECIO
RAQUETA TENIS	DEPORTES	115.2500
MANCUERNAS	DEPORTES	60.0000
PISTOLA OLÍMPICA	DEPORTES	46.7347
BALÓN RUGBY	DEPORTES	111.6440
BALÓN BALONCESTO	DEPORTES	75.2731
BALÓN FÚTBOL	DEPORTES	43.9147
CRONÓMETRO	DEPORTES	439.1764
CAÑA DE PESCA	DEPORTES	270.0000

La RAQUETA TENIS ahora con la actualización se le ha incrementado el precio en 10 euros.

Siguiendo el ejemplo que planteaba al principio en el cual tenemos un grupo de usuarios que están constantemente consultas de las secciones de la tabla PRODUCTOS lo suyo sería crear una segunda vista, tercera vista, etc., que permita ver el resto de secciones.

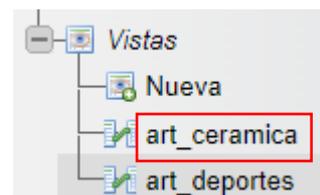
Vamos a crear una segunda vista.

```
1 CREATE VIEW ART_CERAMICA AS
2 SELECT NOMBREARTICULO, SECCION, PRECIO FROM PRODUCTOS
3 WHERE SECCION='CERAMICA'
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Ahora vamos a ver las vistas que tenemos.



Vamos a seleccionar ART_CERAMICA.

NOMBREARTICULO	SECCION	PRECIO
TUBOS	CERÁMICA	168.4253
PLATO DECORATIVO	CERÁMICA	54.0911
JUEGO DE TE	CERÁMICA	43.2728
CENICERO	CERÁMICA	19.7468
MACETA	CERÁMICA	29.0434
JARRA CHINA	CERÁMICA	127.7704

Ahora puedo ver los artículos de DEPORTE por un lado y por otro los artículos de CERAMICA.

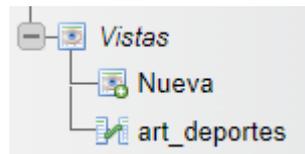
Como podemos eliminar vistas, vamos a eliminar ART_CERAMICA.

```
1 DROP VIEW ART_CERAMICA
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a ver las vistas.



Ya la hemos eliminado.

Algunos gestores de bases de datos soportan las modificaciones de las vistas.

```
1 ALTER VIEW ART_DEPORTES AS
2 SELECT NOMBREARTICULO, SECCION, PAISDEORIGEN FROM PRODUCTOS
3 WHERE PAISDEORIGEN='ESPAÑA'
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

Vamos a mostrar la vista.

NOMBREARTICULO	SECCION	PAÍSDEORIGEN
DESTORNILLADOR	FERRETERÍA	ESPAÑA
CAMISA CABALLERO	CONFECCIÓN	ESPAÑA
MUÑECA ANDADORA	JUGUETERÍA	ESPAÑA
MARTILLO	FERRETERÍA	ESPAÑA

BALÓN FÚTBOL	DEPORTES	ESPAÑA
PISTOLA CON SONIDOS	JUGUETERÍA	ESPAÑA
MACETA	CERÁMICA	ESPAÑA
LIMA GRANDE	FERRETERÍA	ESPAÑA
CINTURÓN DE PIEL	CONFECCIÓN	ESPAÑA
PANTALÓN	<i>NULL</i>	ESPAÑA

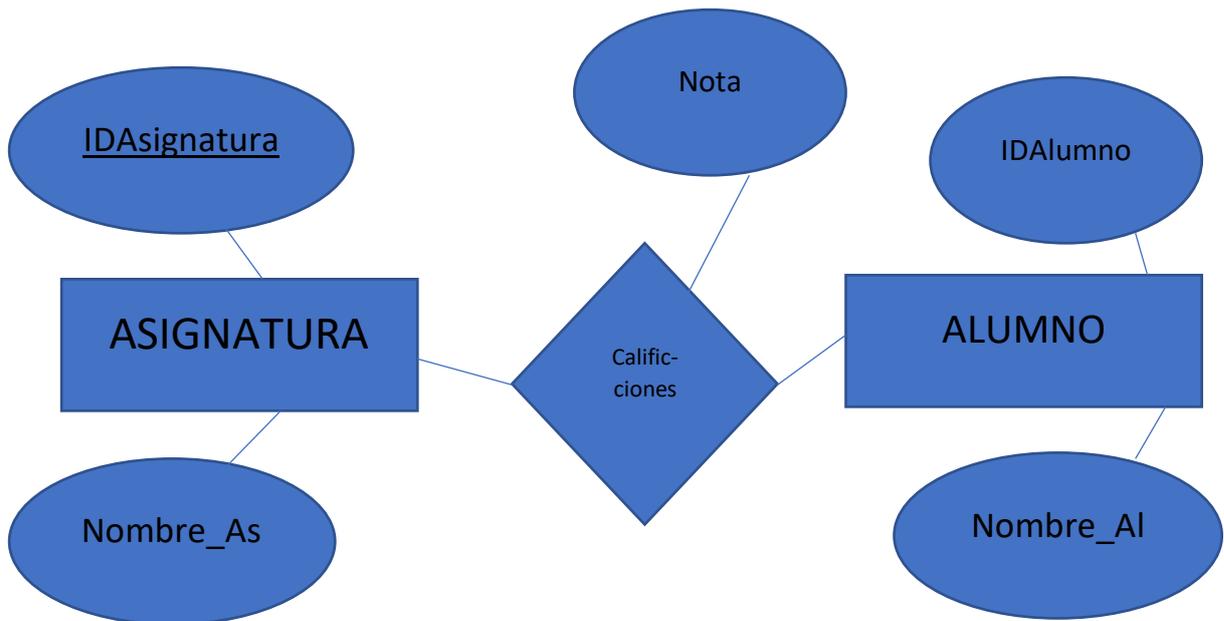
Capítulo 26.- Ejercicio final I (Modelos)

Planteamiento del problema:

En un colegio se requiere tener un registro de calificaciones de los alumnos en sus respectivas asignaturas.

- ALUMNO nombre
- ASIGNATURA nombre

Modelo entidad relación:



Modelo relacional:

ASIGNATURA		
ID_ASIGNATURA (INTEGER)	NOMBRE_AS (VARCHAR)	
1	Matemáticas	
2	Lengua	

CALIFICACIONES		
ID_ASIGNATURA (INTEGER)	ID_ALUMNO (INTEGER)	Notas (INTEGER)

ALUMNO	
ID_ALUMNO (INTEGER)	NOMBRE_AL (VARCHAR)
1	Pedro Martínez
2	Luis Pérez

Capítulo 27.- Ejercicio Final II (Creando la BD)

Creación de base de datos en SQL.

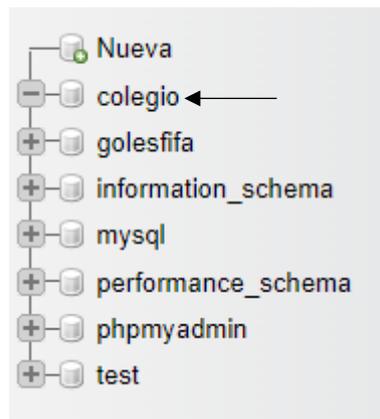
Vamos a crear una base de datos Colegio.

A continuación vamos a crear las tablas ASIGNAURA y ALUMNO.

```
1 CREATE DATABASE Colegio;
```

Le damos a Continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0006 segundos.)



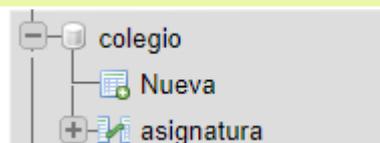
Seleccionamos la base de datos colegio.

A continuación vamos a crear la tabla ASIGNATURA.

```
1 CREATE TABLE ASIGNATURA (ID_ASIGNATURA INTEGER PRIMARY KEY,  
2 NOMBRE_AS VARCHAR(30))
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0006 segundos.)

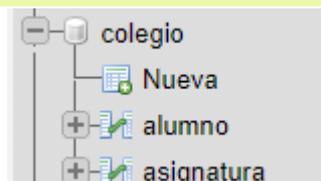


Vamos a crear la tabla ALUMNO.

```
1 CREATE TABLE ALUMNO (ID_ALUMNO INTEGER PRIMARY KEY,  
2 NOMBRE_AL VARCHAR(30));
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)



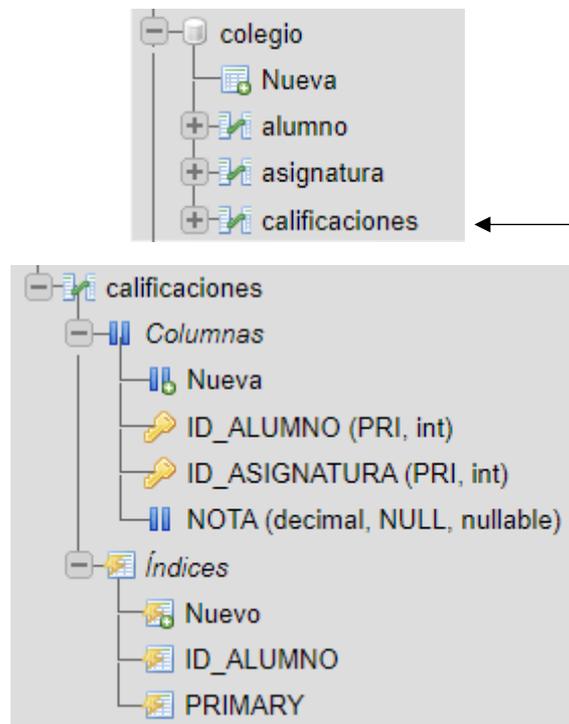
Capítulo 28.- Ejercicio Final III (Creando la tabla CALIFICACIONES)

Vamos a crear la tabla CALIFICACIONES que va a estar relacionada con la tabla ALUMNOS y ASIGNATURAS.

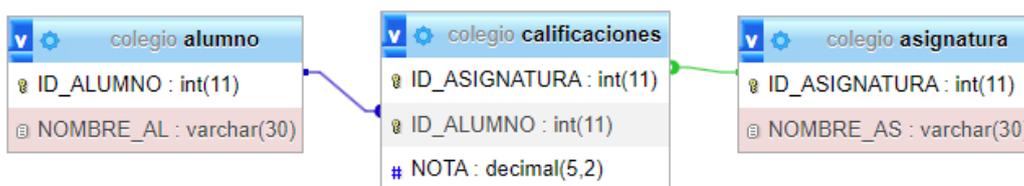
```
1 # Crear tabla CALIFICACIONES
2 CREATE TABLE CALIFICACIONES(
3     ID_ASIGNATURA INTEGER NOT NULL, # Este campo ENTERO no puede estar vacío.
4     ID_ALUMNO INTEGER NOT NULL, # Este campo ENTERO no puede estar vacío.
5     NOTA DECIMAL(5,2), # Campo que guarda las calificaciones.
6     PRIMARY KEY(ID_ASIGNATURA, ID_ALUMNO), # La clave es la suma de
7         # ID_ASIGNATURA + ID_ALUMNO
8     # Vamos a relacionar el campo ID_ASIGNATURA con la tabla ASIGNATURA.
9     FOREIGN KEY (ID_ASIGNATURA) REFERENCES ASIGNATURA(ID_ASIGNATURA),
10    # Vamos a relacionar el campo ID_ALUMNO con la tabla ALUMNO.
11    FOREIGN KEY (ID_ALUMNO) REFERENCES ALUMNO(ID_ALUMNO));
```

Le damos a continuar.

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)



Si vamos a la pestaña diseñador.



Capítulo 29.- Ejercicio Final IV (Insertando Datos)

En este capítulo vamos a añadir registros en las tablas ALUMNOS y ASIGNATURAS.

Tabla ALUMNOS:

```
1 INSERT INTO ALUMNO VALUES(1,'Juan Fernández');
2 INSERT INTO ALUMNO VALUES(2,'Lidia Soriano');
3 INSERT INTO ALUMNO VALUES(3,'Pedro Martínez');
4 INSERT INTO ALUMNO VALUES(4,'Ana Durán');
5 INSERT INTO ALUMNO VALUES(5,'David Zamora');
```

Le damos a continuar.

✓ 1 fila insertada. (La consulta tardó 0,0003 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0011 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0002 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0002 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0002 segundos.)

Vamos a ver el contenido de la tabla.

```
1 SELECT * FROM ALUMNO;
```

Le damos a continuar.

ID_ALUMNO	NOMBRE_AL
1	Juan Fernández
2	Lidia Soriano
3	Pedro Martínez
4	Ana Durán
5	David Zamora

Añadimos a la tabla ASIGNATURA

```
1 INSERT INTO ASIGNATURA VALUES(1,'Matemáticas');
2 INSERT INTO ASIGNATURA VALUES(2,'Lengua');
3 INSERT INTO ASIGNATURA VALUES(3,'Humanidades');
4 INSERT INTO ASIGNATURA VALUES(4,'Inglés');
5 INSERT INTO ASIGNATURA VALUES(5,'Robótica');
```

Le damos a continuar.

✓ 1 fila insertada. (La consulta tardó 0,0003 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0009 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0003 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0003 segundos.)

✓ 1 fila insertada. (La consulta tardó 0,0002 segundos.)

Vamos a ver el contenido de la tabla.

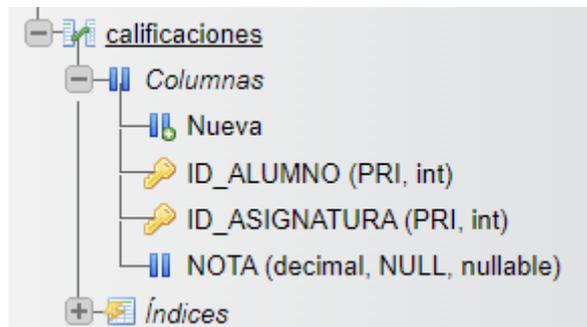
```
1 SELECT * FROM ASIGNATURA;
```

Le damos a continuar.

ID_ASIGNATURA	NOMBRE_AS
1	Matemáticas
2	Lengua
3	Humanidades
4	Inglés
5	Robótica

Capítulo 30.- Ejercicio Final V (Insertando datos tabla CALIFICACIONES)

En este capítulo vamos a añadir registros en la tabla CALIFICACIONES.

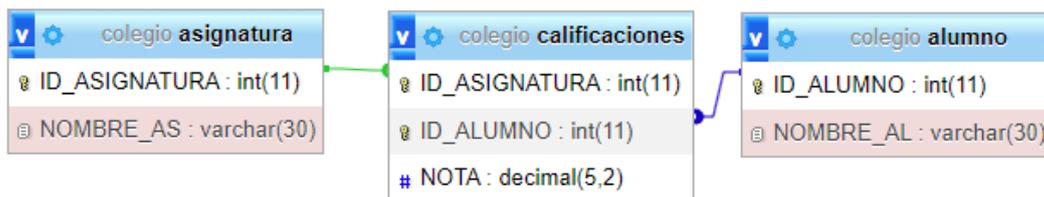


Esta tabla es una tabla compuesta de dos campos ID-ALUMNO y ID_ASIGNATURA que son los dos campos que van a componer la llave de esta tabla.

Tenemos una llave compuesta por dos campos ID-ALUMNO y ID_ASIGNATURA.

¿Cómo vamos a añadir los registros en esta tabla?

Tenemos que tener en cuenta:



En la tabla CALIFICACIONES está relacionada con la tabla ASIGNATURA y ALUMNO por sus respectivos ID, esto significa que si introducimos en la tabla CALIFICACIONES registros donde el ID_ASIGNATURA no está registrado en la tabla ASIGNATURA o en el ID_ALUMNO no está registrado en la tabla ALUMNO también nos dará error.

Para insertar registros en esta tabla es interesante tener la relación de ALUMNOS y ASIGNATURAS para agregarlos correctamente.

1 SELECT * FROM ALUMNO;		1 SELECT * FROM ASIGNATURA;	
ID_ALUMNO	NOMBRE_AL	ID_ASIGNATURA	NOMBRE_AS
1	Juan Fernández	1	Matemáticas
2	Lidia Soriano	2	Lengua
3	Pedro Martínez	3	Humanidades
4	Ana Durán	4	Inglés
5	David Zamora	5	Robótica

A partir de ahora ya podemos añadir los registros de la tabla CALIFICACIONES.

```
1 INSERT INTO CALIFICACIONES VALUES(1,1,7.2); # Juan Fernández en Matemáticas
2 INSERT INTO CALIFICACIONES VALUES(1,2,8.7); # Juan Fernández en Lengua
3 INSERT INTO CALIFICACIONES VALUES(1,3,6.6); # Juan Fernández en Humanidades
4 INSERT INTO CALIFICACIONES VALUES(1,4,7.0); # Juan Fernández en Inglés
5 INSERT INTO CALIFICACIONES VALUES(1,5,9.2); # Juan Fernández en Robótica
```

```
1 INSERT INTO CALIFICACIONES VALUES(2,1,6.5); # Lidia Soriano en Matemáticas
2 INSERT INTO CALIFICACIONES VALUES(2,2,9.3); # Lidia Soriano en Lengua
3 INSERT INTO CALIFICACIONES VALUES(2,3,8.0); # Lidia Soriano en Humanidades
4 INSERT INTO CALIFICACIONES VALUES(2,4,10.0); # Lidia Soriano en Inglés
5 INSERT INTO CALIFICACIONES VALUES(2,5,7.9); # Lidia Soriano en Robótica
```

```
1 INSERT INTO CALIFICACIONES VALUES(3,1,4.9); # Pedro Martínez en Matemáticas
2 INSERT INTO CALIFICACIONES VALUES(3,2,6.2); # Pedro Martínez en Lengua
3 INSERT INTO CALIFICACIONES VALUES(3,3,7.3); # Pedro Martínez en Humanidades
4 INSERT INTO CALIFICACIONES VALUES(3,4,6.4); # Pedro Martínez en Inglés
5 INSERT INTO CALIFICACIONES VALUES(3,5,8.2); # Pedro Martínez en Robótica
```

```
1 INSERT INTO CALIFICACIONES VALUES(4,1,9.3); # Ana Durán en Matemáticas
2 INSERT INTO CALIFICACIONES VALUES(4,2,10); # Ana Durán en Lengua
3 INSERT INTO CALIFICACIONES VALUES(4,3,8.5); # Ana Durán en Humanidades
4 INSERT INTO CALIFICACIONES VALUES(4,4,10); # Ana Durán en Inglés
5 INSERT INTO CALIFICACIONES VALUES(4,5,7,4); # Ana Durán en Robótica
```

```
1 INSERT INTO CALIFICACIONES VALUES(5,1,6.1); # David Zamora en Matemáticas
2 INSERT INTO CALIFICACIONES VALUES(5,2,7.2); # David Zamora en Lengua
3 INSERT INTO CALIFICACIONES VALUES(5,3,5.4); # David Zamora en Humanidades
4 INSERT INTO CALIFICACIONES VALUES(5,4,9); # David Zamora en Inglés
5 INSERT INTO CALIFICACIONES VALUES(5,5,10); # David Zamora en Robótica
```

Todo los alumnos ya tienen introducidas sus correspondientes notas.

Ahora queremos ver los registros que hemos añadido en la tabla CALIFICACIONES:

```
1 SELECT * FROM CALIFICACIONES;
```

Le damos a continuar.

ID_ASIGNATURA	ID_ALUMNO	NOTA
1	1	7.20
1	2	8.70
1	3	6.60
1	4	7.00
1	5	9.20

2	1	6.50
2	2	9.30
2	3	8.00
2	4	10.00
2	5	7.90
3	1	4.90
3	2	6.20
3	3	7.30
3	4	6.40
3	5	8.20
4	1	9.30
4	2	10.00
4	3	8.50
4	4	10.00
4	5	7.40
5	1	6.10
5	2	7.20
5	3	5.40
5	4	9.00
5	5	10.00

Capítulo 31.- Ejercicio Final VI (Alias o sobrenombres)

Utilizamos ALIAS para cambiar el nombre de las columnas.

<pre>1 SELECT ID_ALUMNO, NOMBRE_AL 2 FROM ALUMNO;</pre>	<table border="1"><thead><tr><th>ID_ALUMNO</th><th>NOMBRE_AL</th></tr></thead><tbody><tr><td>1</td><td>Juan Fernández</td></tr><tr><td>2</td><td>Lidia Soriano</td></tr><tr><td>3</td><td>Pedro Martínez</td></tr><tr><td>4</td><td>Ana Durán</td></tr><tr><td>5</td><td>David Zamora</td></tr></tbody></table>	ID_ALUMNO	NOMBRE_AL	1	Juan Fernández	2	Lidia Soriano	3	Pedro Martínez	4	Ana Durán	5	David Zamora
ID_ALUMNO	NOMBRE_AL												
1	Juan Fernández												
2	Lidia Soriano												
3	Pedro Martínez												
4	Ana Durán												
5	David Zamora												
<pre>1 SELECT ID_ALUMNO AS ID, 2 NOMBRE_AL FROM ALUMNO;</pre>	<table border="1"><thead><tr><th>ID</th><th>NOMBRE_AL</th></tr></thead><tbody><tr><td>1</td><td>Juan Fernández</td></tr><tr><td>2</td><td>Lidia Soriano</td></tr><tr><td>3</td><td>Pedro Martínez</td></tr><tr><td>4</td><td>Ana Durán</td></tr><tr><td>5</td><td>David Zamora</td></tr></tbody></table>	ID	NOMBRE_AL	1	Juan Fernández	2	Lidia Soriano	3	Pedro Martínez	4	Ana Durán	5	David Zamora
ID	NOMBRE_AL												
1	Juan Fernández												
2	Lidia Soriano												
3	Pedro Martínez												
4	Ana Durán												
5	David Zamora												
<pre>1 SELECT ID_ALUMNO AS ID, 2 NOMBRE_AL AS 'NOMBRE ALUMNO' 3 FROM ALUMNO;</pre>	<table border="1"><thead><tr><th>ID</th><th>NOMBRE ALUMNO</th></tr></thead><tbody><tr><td>1</td><td>Juan Fernández</td></tr><tr><td>2</td><td>Lidia Soriano</td></tr><tr><td>3</td><td>Pedro Martínez</td></tr><tr><td>4</td><td>Ana Durán</td></tr><tr><td>5</td><td>David Zamora</td></tr></tbody></table>	ID	NOMBRE ALUMNO	1	Juan Fernández	2	Lidia Soriano	3	Pedro Martínez	4	Ana Durán	5	David Zamora
ID	NOMBRE ALUMNO												
1	Juan Fernández												
2	Lidia Soriano												
3	Pedro Martínez												
4	Ana Durán												
5	David Zamora												

Esto es muy útil cuando queremos realizar subconsultas dentro de SQL.

Capítulo 32.- Ejercicio Final VII (UNIONES ENTRE TABLAS)

Queremos obtener la calificación de cada alumno por asignatura y al final queremos revisar la nota de Pedro Martínez en Matemáticas.

```
1 SELECT NOMBRE_AL, NOMBRE_AS, NOTA
2 FROM ALUMNO ALUM, ASIGNATURA ASIN, CALIFICACIONES CALI
3 WHERE ALUM.ID_ALUMNO=CALI.ID_ALUMNO
4 AND ASIN.ID_ASIGNATURA=CALI.ID_ASIGNATURA
```

Le damos a continuar.

NOMBRE_AL	NOMBRE_AS	NOTA
Juan Fernández	Matemáticas	7.20
Juan Fernández	Lengua	6.50
Juan Fernández	Humanidades	4.90
Juan Fernández	Inglés	9.30
Juan Fernández	Robótica	6.10
Lidia Soriano	Matemáticas	8.70
Lidia Soriano	Lengua	9.30
Lidia Soriano	Humanidades	6.20
Lidia Soriano	Inglés	10.00
Lidia Soriano	Robótica	7.20
Pedro Martínez	Matemáticas	6.60
Pedro Martínez	Lengua	8.00
Pedro Martínez	Humanidades	7.30
Pedro Martínez	Inglés	8.50
Pedro Martínez	Robótica	5.40
Ana Durán	Matemáticas	7.00
Ana Durán	Lengua	10.00
Ana Durán	Humanidades	6.40
Ana Durán	Inglés	10.00
Ana Durán	Robótica	9.00
David Zamora	Matemáticas	9.20
David Zamora	Lengua	7.90
David Zamora	Humanidades	8.20
David Zamora	Inglés	7.40
David Zamora	Robótica	10.00

Queremos consultar la nota de Pedro Martínez en Matemáticas.

```
1 SELECT NOMBRE_AL, NOMBRE_AS, NOTA
2 FROM ALUMNO ALUM, ASIGNATURA ASIN, CALIFICACIONES CALI
3 WHERE ALUM.ID_ALUMNO=CALI.ID_ALUMNO
4 AND ASIN.ID_ASIGNATURA=CALI.ID_ASIGNATURA
5 AND ALUM.NOMBRE_AL='Pedro Martínez'
6 AND ASIN.NOMBRE_AS='Matemáticas';
```

Le damos a continuar.

NOMBRE_AL	NOMBRE_AS	NOTA
Pedro Martínez	Matemáticas	6.60

EJERCICIO PRÁCTICO:

Te propongo el siguiente ejercicio, siguiendo el ejemplo anterior:

Un una base de datos llamada BIBLIOTECA, tenemos las siguientes tablas

LECTOR → ID_LECTOR, NOMBRE_LE

LIBRO → ID_LIBRO, NOMBRE_LI

LECTURAS → ID_LECTOR, ID_LIBROS, VALORACION, COMENTARIO

Los lectores tienen que hacer una valoración del libro del 1 al 5 y un pequeño comentario.

Tienes que añadir a la tabla LECTOR a 5 personas.

Tienes que añadir a la tabla LIBRO a 10 títulos de libros.

En la tabla LECTURAS tienes que añadir 15 lecturas añadiendo el ID_LECTOR, ID_LIBRO, VALORACIÓN y COMENTARIO.

Al final queremos realizar la consulta de un determinado lector con un determinado libro que valoración le ha dado y el comentario.

Si puedes que sea este resultado:

ID_LECTOR	ID_LIBRO	VALORACION	COMENTARIO
Pedro Martínez	Marea negra	4	Muy interesante

Si has entendido el Ejercicio Final esta practica la vas a realizar perfectamente.