



## UD 03.5. TEORÍA DE BBDD OFIMÁTICAS

TICD  
ACFGS

M<sup>a</sup> Carmen Safont

[mariacarmen.safont@ceedcv.es](mailto:mariacarmen.safont@ceedcv.es)

Carlos Aparicio

[carlos.aparicio@ceedcv.es](mailto:carlos.aparicio@ceedcv.es)

2019/2020

Versión:191129.2322

Adaptación de los apuntes de Carlos Cacho y Raquel Torres

## Licencia



**Reconocimiento - NoComercial - CompartirIgual (by-nc-sa):** No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

## ÍNDICE DE CONTENIDO

<b>1. Tablas, consultas, formularios e informes.....</b>	<b>3</b>
1.1 Tablas.....	3
1.2 Consultas.....	4
1.3 Formularios.....	5
1.4 Informes.....	6
<b>2. Campos clave.....</b>	<b>7</b>
<b>3. Introducción a las relaciones entre tablas.....</b>	<b>9</b>
3.1 Relaciones entre tablas.....	9
3.1.1 ¿Cuales son las ventajas de establecer una relación entre tablas?.....	9
3.1.2 ¿Cómo se establece una relación entre tablas?.....	10
3.1.3 ¿Qué tipo de relaciones existen entre tablas?.....	10
3.2 Integridad referencial.....	12
3.2.1 Actualizar en cascada los campos relacionados.....	12
3.2.2 Eliminar en cascada los campos relacionados.....	12
<b>4. Bibliografía.....</b>	<b>13</b>

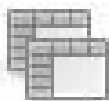
## UD03.5. TEORÍA DE BBDD OFIMÁTICAS

Los programas como el LibreOffice Base gestores de bases de datos facilitan el almacenamiento de la información en estructuras diseñadas por el usuario.

También disponen de herramientas para seleccionar la parte de la información con la que se va a trabajar, es decir, realizar consultas. Y por último, permiten diseñar la apariencia externa de esta información, es decir, formularios e informes.

### 1. TABLAS, CONSULTAS, FORMULARIOS E INFORMES

#### 1.1 Tablas






Las **tablas** de una base de datos conforman la estructura principal de la misma. Podríamos decir que se trata de los cimientos de nuestra estructura.

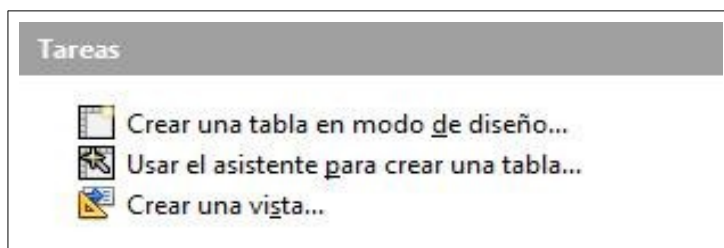
Se puede definir una base de datos como una colección de datos relacionados entre sí. Cada colección de datos relacionados se almacena en una base de datos en forma de tablas, donde dichas tablas guardan información sobre un grupo de cosas con las mismas características o atributos.

Por ejemplo, en una base de datos puedes tener una tabla con información acerca de todos tus libros donde puedes almacenar el título, el nombre del autor, el año de publicación, etc.; es decir, diferentes propiedades o atributos comunes a todos los libros, y, también puedes tener una tabla con toda la información de contacto de tus amigos, por ejemplo, su nombre y apellidos, su número de móvil, su dirección de correo electrónico, etc., pero nunca en una misma tabla se guardará información de cosas con propiedades distintas. Es decir, no puedes mezclar información en una única tabla de tus libros y tus amigos porque la información (los atributos de cada una de las cosas sobre las que guardas información) no es la misma.

Cada uno de los atributos o propiedades de las cosas sobre las que se quiere guardar información son las columnas de la tabla (campos), mientras que cada fila de la tabla representa una cosa real de la que se guarda información (registros o tuplas).

La tabla se puede diseñar usando tres herramientas:

-  Crear una tabla en modo de **diseño**.
-  Usar el **asistente** para crear una tabla
-  Crear una **vista**

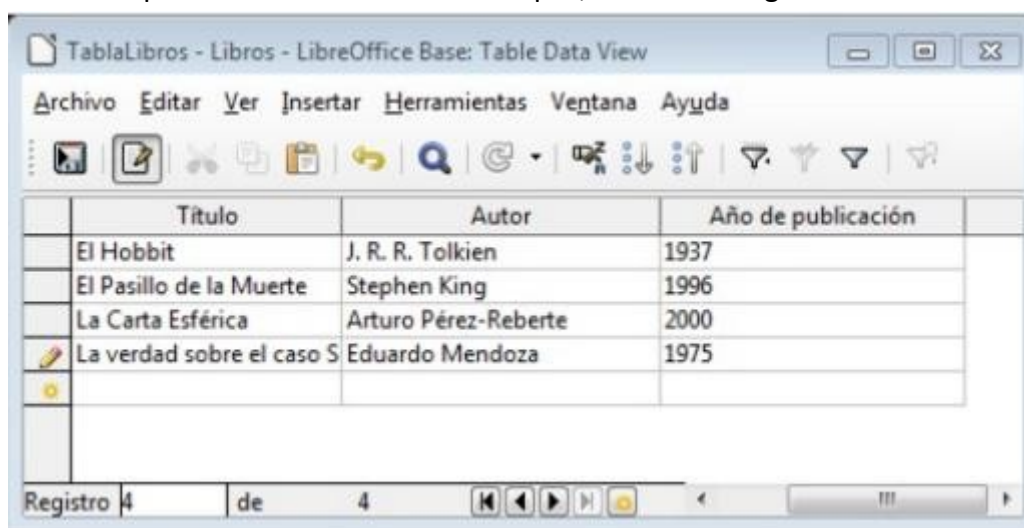


Si se utiliza el **asistente**, la aplicación permite elegir entre una selección de muestras de tablas personales y comerciales que se pueden personalizar para crear una tabla.

El modo de **diseño** y la **vista** son dos maneras diferentes de crear una tabla especificando las tablas, los campos y los tipos de datos. El modo vista es más claro visualmente, y el modo diseño sigue un orden más lógico en la toma de decisiones.

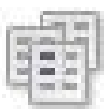
Al final, el resultado es una **tabla** en la que se determinan:

- los campos que tendrá la tabla
- los campos clave (se explica más adelante, aunque ya vimos en el tema anterior el concepto de llave o clave primaria)
- el formato de los campos (si son textos, números, listas ...; la longitud en caracteres de la información que introduciremos en los campos; si son de obligatorio llenado o no ... etc)



	Título	Autor	Año de publicación	
	El Hobbit	J. R. R. Tolkien	1937	
	El Pasillo de la Muerte	Stephen King	1996	
	La Carta Esférica	Arturo Pérez-Reverte	2000	
	La verdad sobre el caso S	Eduardo Mendoza	1975	

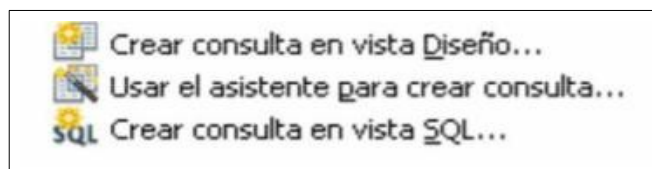
## 1.2 Consultas



Una **consulta** es un objeto de la base de datos que permite seleccionar parte de la información contenida en una o varias tablas para visualizarla en forma de otra tabla. La consulta se realiza definiendo las condiciones que deben cumplir uno o varios campos de dichas tablas.

El resultado de una consulta se ofrece en forma de tabla. Sin embargo, su realización suele estar ligada a la presentación de informes, puesto que permite seleccionar sólo la parte del contenido de la tabla que se va a manipular.

La consulta se puede diseñar usando tres herramientas:



- Crear una consulta en vista diseño.
- Usar el asistente de creación de consultas.
- Crear una consulta en lenguaje SQL

**Crear consulta en vista SQL:** SQL es el lenguaje estándar de creación y manipulación de bases de datos, y requiere su aprendizaje previo a la utilización en la base de datos.

El Asistente para la creación de consultas es una herramienta que facilita la selección de los campos y el establecimiento de las condiciones de la consulta sin necesidad de entrar en el diseño. Parece complicado, pero se trata de un asistente, es decir, es un secretario que va preguntando qué hacer con cada elemento de la consulta.

La creación de consultas en vista diseño consiste en la selección de las tablas, los campos y las condiciones de filtrado y de agrupamiento campo a campo. Al igual que la creación de tablas, requiere un estudio pormenorizado de los condicionantes de la consulta.

El entorno de trabajo se divide en dos zonas: la zona superior, en la que se muestran las tablas seleccionadas, y la zona inferior, en la que se muestran los campos incluidos en la consulta y condiciones que deben cumplir.



El proceso de creación comienza con la selección de las tablas y a continuación, se seleccionan los campos que se visualizarán.

### 1.3 Formularios



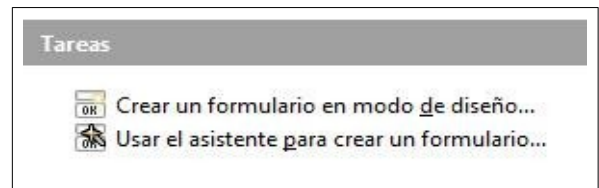
Un formulario es una pantalla de introducción, consulta o modificación de datos en una o varias tablas

La introducción y manipulación de la información en la vista de datos, ya sean en una tabla o en una consulta, es poco clara. Es fácil confundir los registros y cometer fallos al teclear datos. Para evitar estos problemas se usan los formularios.

NumPedido	Destinatario	NumCliente	Destino	FechaPedido	IDPedido	Destinatario	IDCliente	DirecciónDestinatario	CiudadDestinatario
2	empresa 2	32	madrid	22/01/10	1	empresa 1	42	c/ dirección 1	toledo
6	empresa 2	32	madrid	05/02/10					
9	empresa 2	32	madrid	12/02/10					
13	empresa 2	32	madrid	18/02/10					
15	empresa 2	32	madrid	19/02/10					

Al igual que las consultas y las tablas los formularios pueden ser creados en vista diseño o mediante un asistente.

Del mismo modo, el asistente guiará nuestros pasos, mientras que el modo diseño permitirá un diseño totalmente personalizado.

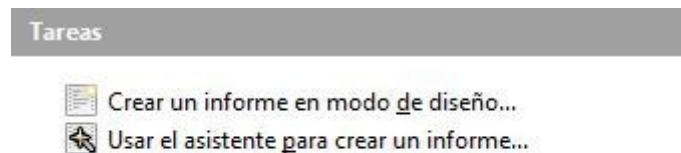


#### 1.4 Informes



Un informe es un objeto de una base de datos que permite la presentación de los datos en la vista de impresión. Igual que un formulario, un informe puede realizarse con los datos de tablas o consultas.

Sin embargo, un informe se ejecuta sobre los datos actuales de la base de datos, siendo, por tanto, un objeto para mostrar datos. No permite la modificación ni la introducción de nuevos registros. Esta es la diferencia con los formularios.



Como en los demás casos, el asistente guiará nuestros pasos, mientras que el modo diseño permitirá o un diseño totalmente personalizado.

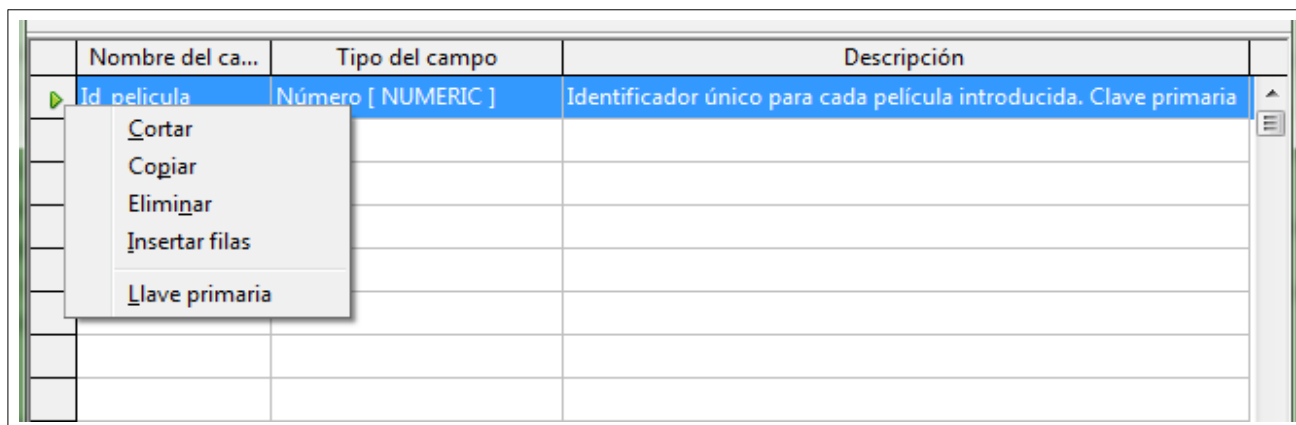
## 2. CAMPOS CLAVE

En cualquier base de datos los registros incluidos en sus diferentes tablas deben estar perfectamente identificados y de esto se encargan las claves o llaves. Trasladando este concepto a la vida real, cada ciudadano tiene un número de DNI, puede haber dos personas con igual nombre e incluso apellidos iguales, pero ambos se diferenciarán por su número de DNI, que es único en "teoría".

Cada tabla debe contener al menos un campo que permita identificar unívocamente cada registro. Este campo puede ser real, como el propio DNI de una persona. O puede ser un valor que nos inventemos para conseguir el mismo fin, como un número que asignemos automáticamente a cada registro, asegurándonos (realmente lo hará Base) de que no se repita. Otra opción es utilizar una combinación de campos para identificar los registros de la tabla. Por ejemplo podríamos definir la combinación de Nombre y Apellidos de nuestra tabla para este fin. Sin embargo, existe la probabilidad de que haya personas con igual nombre y apellidos, por lo que no sería una clave del todo válida.

Imaginemos que tenemos una tabla llamada "Películas" en la que, a priori, no hay ningún campo acerca de una película que podamos utilizar para diferenciar una película de otra ya que, por ejemplo, podemos tener dos películas con el mismo título, o dos películas con el mismo director, o dos películas que se produjeron el mismo año. Para este tipo de tablas se suele definir un campo de tipo numérico (como acabamos de hacer con Id\_pelicula) que debe cumplir dos condiciones: la primera, es que ese campo siempre tiene que tomar un valor para cada fila de la tabla, y, la segunda, que dicho valor no puede repetirse en ninguna fila. Cuando un campo cumple estas dos propiedades (sin nulos y sin repetidos) se le llama Llave Primaria, Clave Primaria o simplemente Clave. Toda tabla debe tener una.

Con el botón derecho del ratón pulsamos sobre la flecha verde y elegimos "Llave primaria" del menú contextual que se nos abre.




	Nombre del ca...	Tipo del campo	Descripción
▶	Id_pelicula	Número [ NUMERIC ]	Identificador único para cada película introducida. Clave primaria

Contextual menu options:

- Cortar
- Copiar
- Eliminar
- Insertar filas
- Llave primaria

Y el campo queda identificado como llave primaria, mostrando el símbolo correspondiente (una llave junto a la flecha verde).

	Nombre del ca...	Tipo del campo	Descripción
	 Id_pelicula	Número [ NUMERIC ]	Identificador único para cada película introducida. Clave primaria

Si pretendiéramos introducir un nuevo registro en una tabla que estamos creando y le pusiéramos la clave primaria de un registro ya existente (como en la imagen, a la que se intenta poner como Id\_pelicula –la clave primaria de la tabla– el número 1, que ya corresponde a otra película), la aplicación nos indica que no es posible)

LibreOffice Base: Table Data View

Archivo Editar Ver Insertar Herramientas Ventana Ayuda

	Id_pelicula	Título	Director	Año	Formato	Vista
	1	Blade Runner	Ridley Scott	01/01/83	DVD	<input checked="" type="checkbox"/>
	2	La Guerra de las Galaxias	George Lucas	01/01/77	VHS	<input checked="" type="checkbox"/>
	3	En Busca del Arca Perdida	Steven Spilverg	01/01/81	DVD	<input checked="" type="checkbox"/>
	4	Milión Dólar Baby	Clint Eastwood	01/01/04	DVD	<input type="checkbox"/>
	5	La Comunidad del Anillo	Peter Jackson	01/01/01	DVD	<input checked="" type="checkbox"/>
	6	Belleza Robada	Bernardo Bertolucci	01/01/96	VHS	<input checked="" type="checkbox"/>
	7	Gladiator	Ridley Scott	01/01/00	DVD	<input checked="" type="checkbox"/>
	8	Rocky	John G. Avidsen	01/01/76	VHS	<input checked="" type="checkbox"/>
	9	Los Lunes al Sol	Fernando León	01/01/02	DVD	<input type="checkbox"/>
	10	Mar Adentro	Alejandro Amenábar	01/01/04	DVD	<input type="checkbox"/>
	1	Big Fish	Tim Burton	01/01/05	CD	<input type="checkbox"/>

Registro 11

LibreOffice Base

Error al escribir el registro actual de datos.

Violation of unique constraint SYS\_PK\_47: duplicate value(s) for column(s) "Id\_pelicula" in statement [UPDATE "Películas" SET "Id\_pelicula" = ? WHERE "Id\_pelicula" = ?]

Aceptar Más



### 3. INTRODUCCIÓN A LAS RELACIONES ENTRE TABLAS

#### 3.1 Relaciones entre tablas

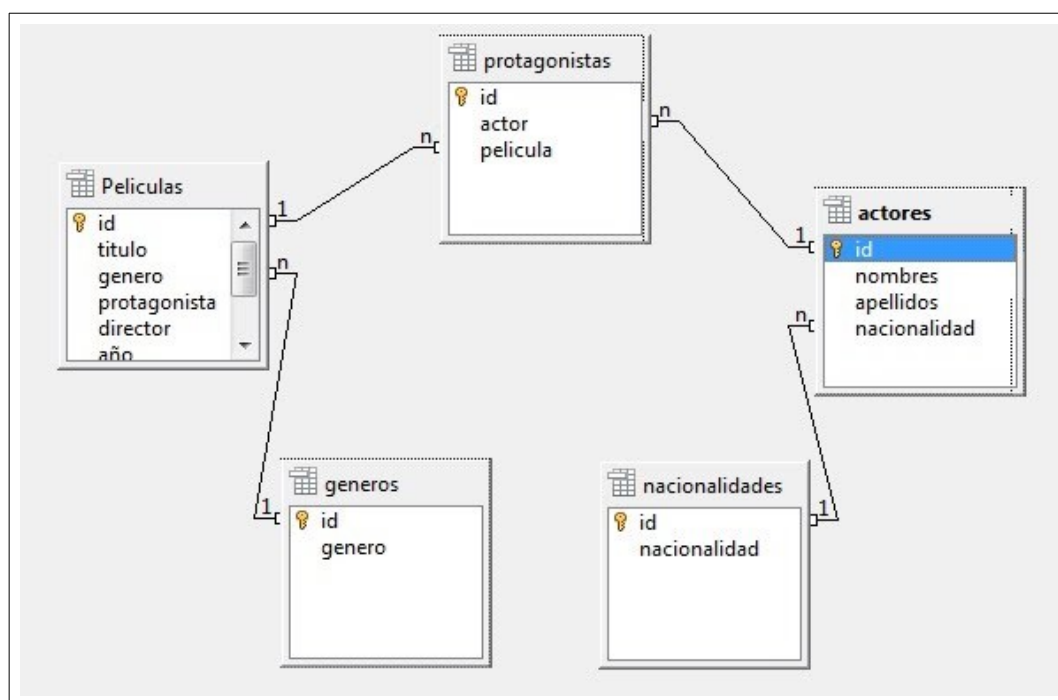
En una base de datos tenemos la información que deseamos almacenar en varias tablas relacionadas entre ellas en lugar de en una única tabla enorme con toda la información (que sería lo que tendríamos si decidiéramos introducir la información en una hoja de cálculo, en lugar de en una base de datos).

##### 3.1.1 ¿Cuales son las ventajas de establecer una relación entre tablas?

Imaginemos que tenemos una base de datos con las películas de nuestra colección. Si, además del título de la película, el director y la fecha de estreno queremos guardar el género al que pertenece la película, podemos añadir un nuevo campo denominado “Género”. Eso nos obligaría a teclear el género en cada una de nuestras películas, que es pesado. Además, en el momento que tecleáramos mal el género, crearíamos un nuevo género, creando confusión. Si, por algún motivo, en algún momento queremos cambiar la denominación al género (por ejemplo, decidimos que a las películas que catalogamos como “Románticas” deseamos cambiarles la denominación a “Drama”, deberemos cambiarlo en cada una de nuestras películas una a una.

Otra ventaja puede verse con el siguiente ejemplo. Si deseamos incluir en nuestra base de datos el nombre del protagonista de la película y su nacionalidad, deberemos crear dos campos nuevos en nuestra tabla “Películas”. Imaginemos ahora que algunas de nuestras películas tienen más de un protagonista, no tendremos más remedio que elegir sólo uno, o crear dos campos más, con lo que nuestra tabla cada vez va haciéndose más incómoda de manejar y de visualizar en pantalla.

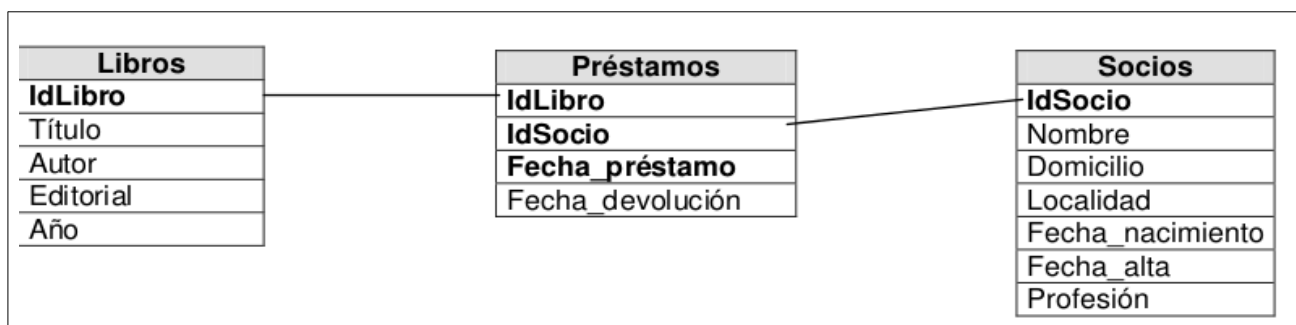
Por tanto, parece más recomendable dejar la tabla “Películas” tal como estaba y crear por otro lado la tabla “Género” y la tabla “Protagonistas”, intentando indicar de alguna manera que van a existir relaciones entre esas tablas.



### 3.1.2 ¿Cómo se establece una relación entre tablas?

Cuando se quiere mostrar información (mediante una consulta o un informe) de dos tablas distintas es necesario previamente crear una relación entre ellas.

Pero dos tablas sólo se podrán relacionar si existe un campo común entre ambas.



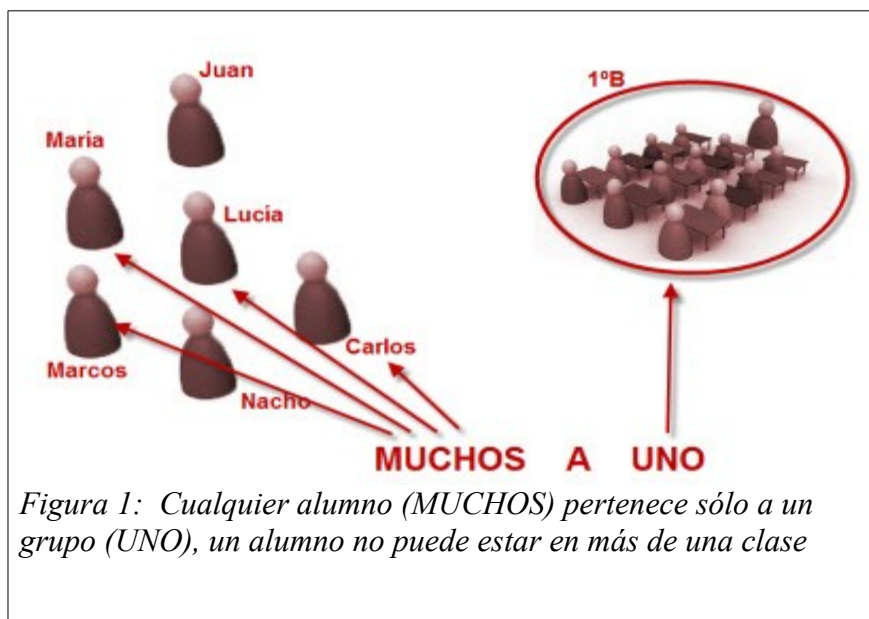
En el ejemplo anterior se pueden establecer dos relaciones:

- Entre la tabla Libros y la tabla Préstamos, ya que tienen un campo común: los valores del campo IdLibro de la tabla Préstamos hacen referencia a los valores del campo IdLibro de la tabla Libros.
- Entre la tabla Socios y la tabla Préstamos: los valores del campo IdSocio de la tabla Préstamos hacen referencia a los valores del campo IdSocio de la tabla Socios.

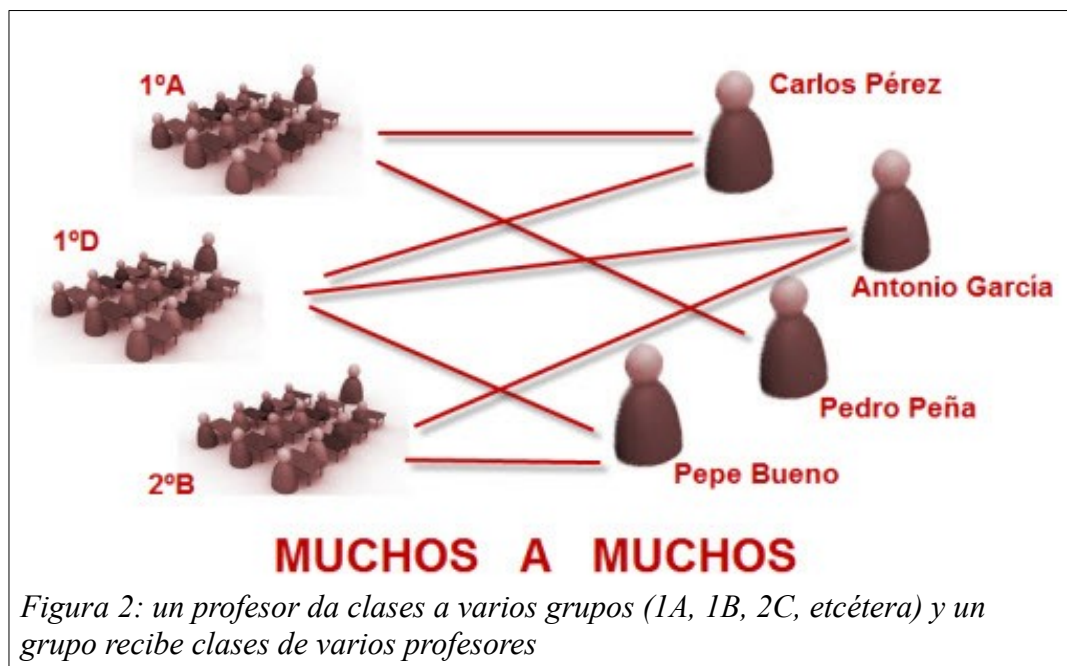
### 3.1.3 ¿Qué tipo de relaciones existen entre tablas?

Existen tres tipos básicos de relaciones entre tablas:

- **Uno a muchos.** Este tipo se da cuando una fila de la primera tabla puede estar relacionada con muchas filas de la segunda tabla, pero una fila de la segunda sólo está relacionada con una de la primera. Es decir, si consideramos que una película sólo puede tener un único género, éste sería el tipo de relación que existe entre la tabla Genero y la tabla Peliculas ya que, por ejemplo, el género Drama tiene varias películas relacionadas que son de dicho género pero dada una película sólo está relacionada con un género.



- **Muchos a muchos.** Esta clase de relación ocurre cuando una fila de la primera tabla puede estar relacionada con muchas filas de la segunda tabla y una fila de la segunda tabla puede estarlo con muchas filas de la primera. Un ejemplo de este tipo lo tenemos en la relación entre la tabla Películas y la tabla Interpretes porque, dada una película en particular, puede tener muchos intérpretes, y, viceversa, dado un intérprete, este puede haber intervenido en muchas películas.

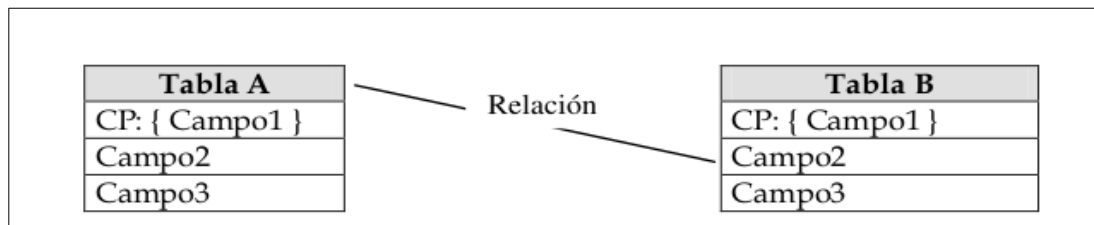


- **Uno a uno.** Este tipo de relación aparece con menos frecuencia y sucede cuando una fila de la primera tabla sólo puede estar relacionada con una fila de la segunda y una fila de la segunda tabla sólo puede estar relacionada con una de la primera. Un ejemplo de este tipo de relaciones podría ser entre una tabla con países y otra con jefes de gobierno, dado que, normalmente, un país sólo tiene un jefe de gobierno y un jefe de gobierno lo es sólo de un país.



### 3.2 Integridad referencial

La integridad referencial es una restricción que se define sobre una relación. En LibreOffice Base todas las relaciones llevan incorporada la integridad referencial.



Los valores del Campo3 de la Tabla B tendrán que hacer referencia a valores que ya existan en el Campo1 de la Tabla A. En caso de que no haga referencia a un valor ya existente se impedirá la actualización. Esto implica que los registros de la Tabla A tendrán que ser introducidos antes que los registros de la Tabla B. De los dos campos relacionados, el campo que es clave primaria es el que contiene los valores a los que tienen que hacer referencia los campos relacionados de otras tablas.

Por ejemplo, al marcar la casilla de Exigir integridad referencial en la relación entre la tabla Libros y la tabla Préstamos, estamos obligando al usuario a que cuando introduzca un nuevo registro en la tabla Préstamos, el campo Libro tendrá que contener el código de un libro que ya exista en la tabla Libros. En caso de que introduzcamos un código que no exista en la tabla Libros se mostrará un mensaje de error y se impedirá la actualización.

La integridad referencial es un mecanismo que nos permite asegurar que los valores que estamos introduciendo son íntegros con los valores que ya existen. En el ejemplo anterior, nos aseguramos que al dar de alta un préstamo el código del libro introducido hace referencia a un libro que realmente exista.

#### 3.2.1 Actualizar en cascada los campos relacionados

Al cambiar el valor del Campo1 en un registro de la tabla A se actualizarán automáticamente en la tabla B los valores del Campo3 que hicieran referencia a ese valor.

Tabla Libros	
IdLibro	Título
10	El nombre de la rosa
20	El médico
30	El mundo de Sofía

Tabla Préstamos		
IdLibro	IdSocio	Fecha préstamo
10	105	21/10/2004
20	105	21/10/2004
20	107	29/11/2004

Si cambiamos el código del libro El médico del 20 al 25, se actualizará en cascada los dos préstamos que hay con el código 20 (los de los socios 105 y 107) al nuevo valor 25.

#### 3.2.2 Eliminar en cascada los campos relacionados

Al eliminar un registro en la Tabla A se eliminarán automáticamente en la Tabla B todos los registros con los que estuviera relacionado.

En el ejemplo anterior, si borramos el libro El médico también se borrarán los dos préstamos relacionados.

#### 4. BIBLIOGRAFÍA

- [http://www.zenodoto.com/gestiondocs/cursosacademicos/manualesrecursos/materialesformaciondocenteTIC/TIC7\\_BASE/indice.htm](http://www.zenodoto.com/gestiondocs/cursosacademicos/manualesrecursos/materialesformaciondocenteTIC/TIC7_BASE/indice.htm)
- [http://www.aulapc.es/ofimatica\\_acces\\_relaciones.html](http://www.aulapc.es/ofimatica_acces_relaciones.html)
- [https://alfabetizaciondigital.fundacionesplai.org/#cm\\_submenu\\_1](https://alfabetizaciondigital.fundacionesplai.org/#cm_submenu_1)
-