

TEC GURUS

www.tecgurus.net



Somos y formamos expertos en T.I.

Índice

- Bases de datos
 - [Definición](#)
 - [Tablas](#)
 - [Columnas](#)
 - [Relaciones](#)
 - [Tipos de relaciones](#)
- UML database diagram

BASES DE DATOS



[This Photo](#) by Unknown Author is licensed under [CC BY](#)

- Una base de datos es una recopilación de información que se organiza para que se pueda acceder, administrar y actualizar fácilmente.
- Los datos se organizan en filas, columnas y tablas, y se indexan para que sea más fácil encontrar información relevante. Los datos se actualizan, expanden y eliminan a medida que se agrega nueva información. Las bases de datos procesan cargas de trabajo para crearse y actualizarse, consultar los datos que contienen y ejecutar aplicaciones en su contra.

<https://searchsqlserver.techtarget.com/definition/database>

TABLAS

Una base de datos consta de una o más tablas. Cada tabla está formada por filas y columnas. Si piensa en una tabla como una cuadrícula, la columna va de izquierda a derecha a través de la cuadrícula y cada entrada de datos se muestra como una fila.

Cada fila en una relación es identificada de manera única por una clave primaria. Esto puede ser por uno o más conjuntos de valores de columna. En la mayoría de los escenarios, es una sola columna, como employeeID.

Cada tabla relacional tiene una clave primaria. Su propósito es identificar de forma única cada fila en la base de datos. No hay dos filas que puedan tener el mismo valor de clave principal. El resultado práctico de esto es que puede seleccionar cada fila solo al conocer su clave principal.

<https://www.essentialsql.com/what-is-a-database-table/>

COLUMNAS

Las columnas se definen para contener un tipo específico de datos, como fechas, datos numéricos o textuales. En la más simple de las definiciones, una columna se define por su nombre y tipo de datos. El nombre se usa en las sentencias de SQL al seleccionar y ordenar datos, y el tipo de datos se usa para validar la información almacenada.

Por lo tanto, una columna `DateOfBirth` definida como una fecha, se puede referir en una cláusula *order by*

`ORDER BY DateOfBirth`

Y, si intenta agregar un valor de "Hello Kitty" a la columna, como parte de su validación, reconocerá que no es una fecha y la rechazará.

Los nombres de las columnas no se pueden duplicar en una tabla. Entonces, tener dos columnas de "nombre" es un no, no. Aunque podría tener dos columnas de "nombre", como `nombre1` y `nombre2`, aprenderá más adelante, que esto está mal visto, ya que rompe la forma normal (lo explico en otra publicación).

<https://www.essentialsql.com/what-is-a-database-table/>

RELACIONES

Al crear una base de datos, el sentido común dicta que usemos tablas separadas para diferentes tipos de entidades.

Algunos ejemplos son: clientes, pedidos, artículos, mensajes, etc. Pero también necesitamos tener relaciones entre estas tablas.

Por ejemplo, los clientes hacen pedidos y los pedidos contienen artículos.

Estas relaciones deben estar representadas en la base de datos. Además, al obtener datos con SQL, necesitamos usar ciertos tipos de consultas de JOIN para obtener lo que necesitamos.

<https://code.tutsplus.com/articles/sql-for-beginners-part-3-database-relationships--net-8561>

TIPOS DE RELACIONES

- Relaciones Uno a Uno
- Relaciones uno a muchos y muchos a uno
- Muchas a muchas relaciones
- Relaciones de autorreferencia

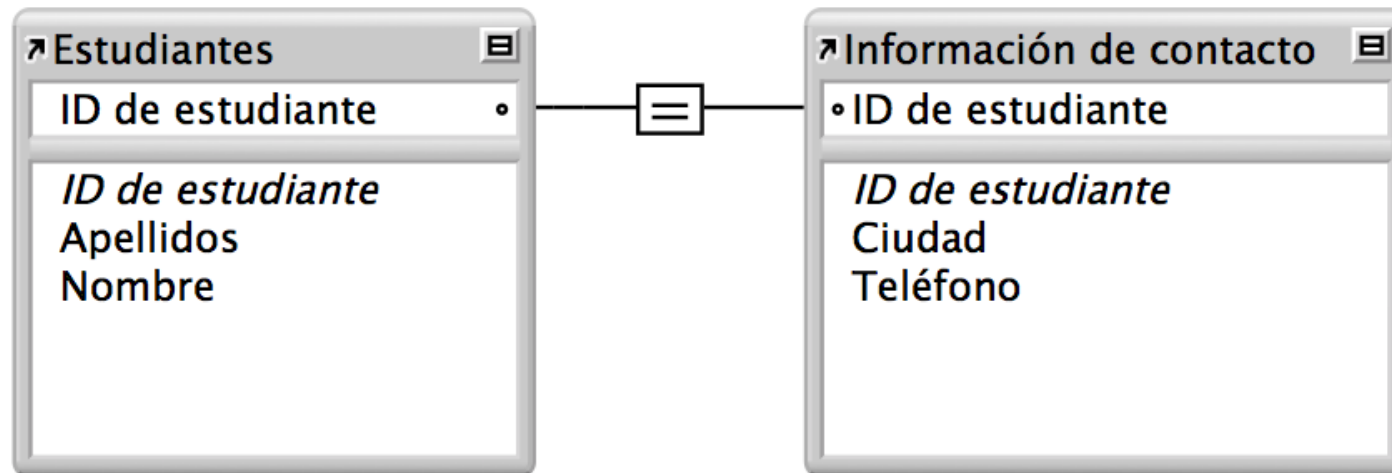
<https://code.tutsplus.com/articles/sql-for-beginners-part-3-database-relationships--net-8561>

RELACIONES UNO A UNO

En una relación de uno a uno, un registro de una tabla se asocia a uno y solo un registro de otra tabla. Por ejemplo, en una base de datos de un centro educativo, cada alumno tiene solamente un ID de estudiante, y cada ID de estudiante se asigna solo a una persona.

Una relación de uno a uno presenta el siguiente aspecto en el gráfico de relaciones:

https://fmhelp.filemaker.com/help/17/fmp/es/index.html#page/FMP_Help/one-to-one-relationships.html

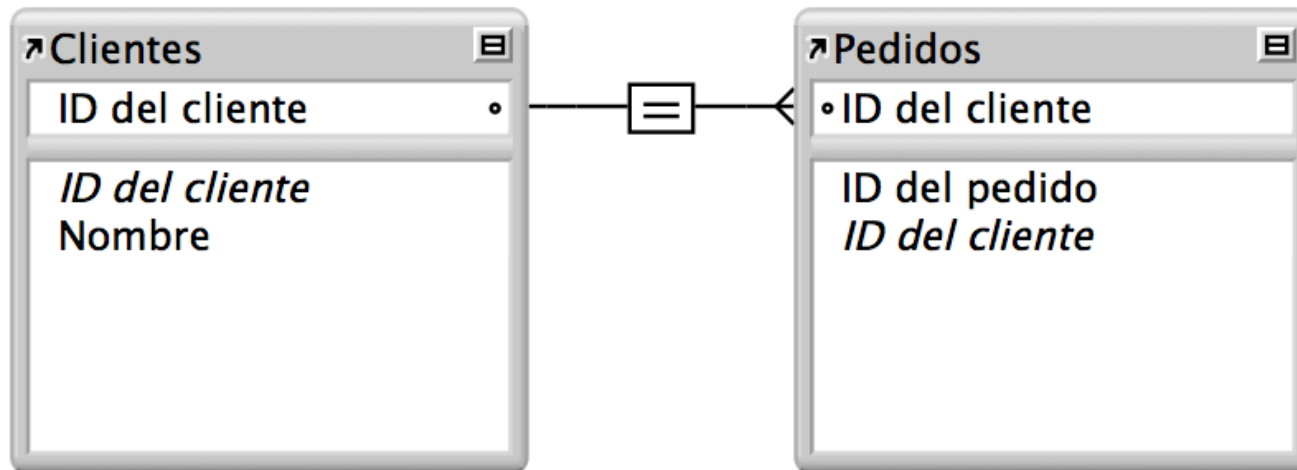


RELACIONES UNO A MUCHOS

En una relación de uno a muchos, un registro de una tabla se puede asociar a uno o varios registros de otra tabla.. Por ejemplo, cada cliente puede tener varios pedidos de ventas.

Una relación de uno a muchos presenta el siguiente aspecto en el gráfico de relaciones:

https://fmhelp.filemaker.com/help/17/fmp/es/index.html#page/FMP_Help/one-to-one-relationships.html



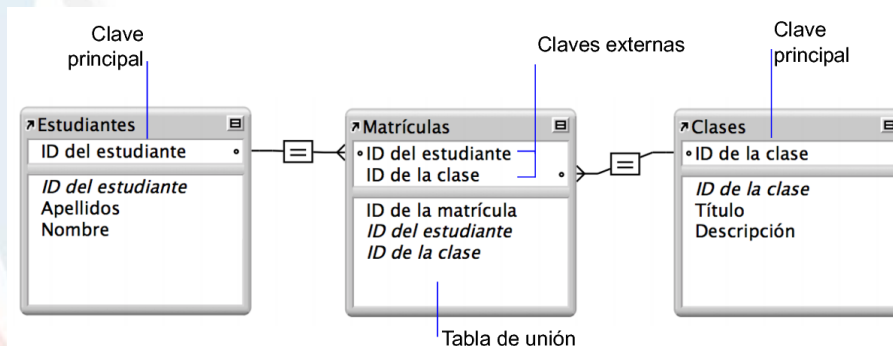
RELACIONES MUCHOS A MUCHOS

Una *relación de muchos a muchos* se produce cuando varios registros de una tabla se asocian a varios registros de otra tabla.

Por lo general, los sistemas de bases de datos relacionales no permiten implementar una relación directa de muchos a muchos entre dos tablas.

Se hace mediante el uso de una tercera tabla denominada *tabla de unión*. Cada registro de una tabla de unión incluye un campo de coincidencia que contiene el valor de las claves principales de las dos tablas que se unen. (En la tabla de unión, estos campos de coincidencia son claves externas). Estos campos de clave externa se rellenan con datos, ya que los registros de la tabla de unión se crean desde cualquiera de las tablas que se unen.

https://fmhelp.filemaker.com/help/17/fmp/es/index.html#page/FMP_Help/one-to-one-relationships.html



RELACIONES DE AUTORREFERENCIA O AUTOUNIÓN

Una *relación de autounión* (o *autounión*) es una relación en la que los dos campos de coincidencia se definen en la misma tabla. FileMaker Pro Advanced genera una segunda ocurrencia de la tabla en la que se basará la relación en el gráfico de relaciones. Esto evita que las relaciones formen un ciclo, que impediría realizar una evaluación.

Por ejemplo, utilice una autounión en un portal de una presentación de la tabla actual para mostrar un subconjunto de datos que se encuentre en esa tabla como, por ejemplo, un subconjunto con todos los empleados de cada director. En este ejemplo, utilizará una tabla denominada Empleados.

https://fmhelp.filemaker.com/help/17/fmp/es/index.html#page/FMP_Help/one-to-one-relationships.html

Usuario
ID_User
ID_UpdatedBy

EJERCICIOS

EJERCICIO 1

Se quiere diseñar una base de datos relacional para almacenar información sobre los asuntos que lleva un gabinete de abogados. Cada asunto tiene un número de expediente que lo identifica, y corresponde a un solo cliente. Del asunto se debe almacenar el período (fecha de inicio y fecha de archivo o finalización), su estado (en trámite, archivado, etc.), así como los datos personales del cliente al que pertenece (DNI, nombre, dirección, etc.). Algunos asuntos son llevados por uno o varios procuradores, de los que nos interesa también los datos personales.

EJERCICIO 2

Se quiere diseñar una base de datos relacional que almacene información relativa a los zoos existentes en el mundo, así como las especies animales que éstos albergan. De cada zoo se conoce el nombre, ciudad y país donde se encuentra, tamaño (en m²) y presupuesto anual. De cada especie animal se almacena el nombre vulgar y nombre científico, familia a la que pertenece y si se encuentra en peligro de extinción. Además, se debe guardar información sobre cada animal que los zoos poseen, como su número de identificación, especie, sexo, año de nacimiento, país de origen y continente.

<http://www3.uji.es/~mmarques/f47/teoria/dis02.pdf>

Ing. Jeniffer Ramirez

