

Referencia Documental.

La información de este artículo fue localizada en la siguiente dirección web: <ms-help://MS.MSACCESS.12.3082/MSACCESS/content/HA01224247.htm#Terms>.

La cual puedes consultar para ampliar la información que aquí se está proporcionando.

Conceptos básicos del diseño de una base de datos

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Puesto que un diseño correcto es esencial para lograr los objetivos fijados para la base de datos, parece lógico emplear el tiempo que sea necesario en aprender los principios de un buen diseño ya que, en ese caso, es mucho más probable que la base de datos termine adaptándose a sus necesidades y pueda modificarse fácilmente.

En este artículo se proporcionan instrucciones para preparar una base de datos. Aprenderá a decidir qué información necesita, a dividir la información en las tablas y columnas adecuadas y a relacionar las tablas entre sí. Debe leer este artículo antes de crear la primera base de datos.

En este artículo

↴ [Algunos términos sobre bases de datos que debe conocer](#)

↴ [¿Qué es un buen diseño de base de datos?](#)

↴ [El proceso de diseño](#)

↴ [Determinar la finalidad de la base de datos](#)

↴ [Buscar y organizar la información necesaria](#)

↴ [Dividir la información en tablas](#)

↴ [Convertir los elementos de información en columnas](#)

↴ [Especificar claves principales](#)

↴ [Crear relaciones entre las tablas](#)

↴ [Ajustar el diseño](#)

↴ [Aplicar las reglas de normalización](#)

↴ [Para obtener más información](#)

Algunos términos sobre bases de datos que debe conocer

Microsoft Office Access 2007 organiza la información en **tablas**, que son listas y columnas similares a las de los libros contables o a las de las hojas de cálculo de Microsoft Office Excel 2007. Una base de datos simple puede que sólo contenga una tabla, pero la mayoría de las bases de datos necesitan varias tablas. Por ejemplo, podría tener una tabla con información sobre productos, otra con información sobre pedidos y una tercera con información sobre clientes.

The image shows three overlapping screenshots of Microsoft Access tables. The top window is titled 'Productos' and contains a table with columns 'Id. de pro...', 'Nombre de pr...', and 'Proveedor'. The middle window is titled 'Clientes' and contains a table with columns 'Nombre de la compañía' and 'Nombre del ...'. The bottom window is titled 'Pedidos' and contains a table with columns 'Id. de p...', 'Cliente', and 'Empleado'. The 'Pedidos' table is currently selected and shows a status bar at the bottom indicating 'Registro 1 de 830' and 'Sin filtro'.

Id. de pro...	Nombre de pr...	Proveedor
1	Chai	Exotic Liquids
2	Chang	Exotic Liquids

Nombre de la compañía	Nombre del ...
Alfreds Futterkiste	Maria Anders
Ana Trujillo Emparedados y helados	Ana Trujillo

Id. de p...	Cliente	Empleado
10248	Wilman Kala	Buchanan, Ste
10249	Tradição Hipermercados	Suyama, Mich
10250	Hanari Carnes	Peacock, Marg

Cada fila recibe también el nombre de **registro** y cada columna se denomina también **campo**. Un registro es una forma lógica y coherente de combinar información sobre alguna cosa. Un campo es un elemento único de información: un tipo de elemento que aparece en cada registro. En la tabla Products (Productos), por ejemplo, cada fila o registro contendría información sobre un producto, y cada columna contendría algún dato sobre ese producto, como su nombre o el precio.


[Volver al principio](#)

¿Qué es un buen diseño de base de datos?

El proceso de diseño de una base de datos se guía por algunos principios. El primero de ellos es que se debe evitar la información duplicada o, lo que es lo mismo, los datos redundantes, porque malgastan el espacio y aumentan la probabilidad de que se produzcan errores e incoherencias. El segundo principio es que es importante que la información sea correcta y completa. Si la base de datos contiene información incorrecta, los informes que recogen información de la base de datos contendrán también información incorrecta y, por tanto, las decisiones que tome a partir de esos informes estarán mal fundamentadas.

Un buen diseño de base de datos es, por tanto, aquél que:

- ↓ Divide la información en tablas basadas en temas para reducir los datos redundantes.
- ↓ Proporciona a Access la información necesaria para reunir la información de las tablas cuando así se precise.
- ↓ Ayuda a garantizar la exactitud e integridad de la información.
- ↓ Satisface las necesidades de procesamiento de los datos y de generación de informes.

 [Volver al principio](#)

El proceso de diseño

El proceso de diseño consta de los pasos siguientes:

↓ **Determinar la finalidad de la base de datos**

Esto le ayudará a estar preparado para los demás pasos.

↓ **Buscar y organizar la información necesaria**

Reúna todos los tipos de información que desee registrar en la base de datos, como los nombres de productos o los números de pedidos.

↓ **Dividir la información en tablas**

Divida los elementos de información en entidades o temas principales, como Productos o Pedidos. Cada tema pasará a ser una tabla.

↓ **Convertir los elementos de información en columnas**

Decida qué información desea almacenar en cada tabla. Cada elemento se convertirá en un campo y se mostrará como una columna en la tabla. Por ejemplo, una tabla Empleados podría incluir campos como Apellido y Fecha de contratación.

↓ **Especificar claves principales**

Elija la clave principal de cada tabla. La clave principal es una columna que se utiliza para identificar inequívocamente cada fila, como Id. de producto o Id. de pedido.

↓ **Definir relaciones entre las tablas**

Examine cada tabla y decida cómo se relacionan los datos de una tabla con las demás tablas. Agregue campos a las tablas o cree nuevas tablas para clarificar las relaciones según sea necesario.

↓ **Ajustar el diseño**

Analice el diseño para detectar errores. Cree las tablas y agregue algunos registros con datos de ejemplo. Compruebe si puede obtener los resultados previstos de las tablas. Realice los ajustes necesarios en el diseño.

↓ **Aplicar las reglas de normalización**

Aplique reglas de normalización de los datos para comprobar si las tablas están estructuradas correctamente. Realice los ajustes necesarios en las tablas.

 [Volver al principio](#)

Determinar la finalidad de la base de datos

Es conveniente plasmar en papel el propósito de la base de datos: cómo piensa utilizarla y quién va a utilizarla. Para una pequeña base de datos de un negocio particular, por ejemplo, podría escribir algo tan

simple como "La base de datos de clientes contiene una lista de información de los clientes para el envío masivo de correo y la generación de informes". Si la base de datos es más compleja o la utilizan muchas personas, como ocurre normalmente en un entorno corporativo, la finalidad podría definirse fácilmente en uno o varios párrafos y debería incluir cuándo y cómo va a utilizar cada persona la base de datos. La idea es desarrollar una declaración de intenciones bien definida que sirva de referencia durante todo el proceso de diseño. Esta declaración de intenciones le permitirá centrarse en los objetivos a la hora de tomar decisiones.

[📄 Volver al principio](#)

Buscar y organizar la información necesaria

Para buscar y organizar la información necesaria, empiece con la información existente. Por ejemplo, si registra los pedidos de compra en un libro contable o guarda la información de los clientes en formularios en papel en un archivador, puede reunir esos documentos y enumerar cada tipo de información que contienen (por ejemplo, cada casilla de un formulario). Si no dispone de formularios, imagine que tiene que diseñar uno para registrar la información de los clientes. ¿Qué información incluiría en el formulario? ¿Qué casillas crearía? Identifique cada uno de estos elementos y cree un listado. Suponga, por ejemplo, que guarda la lista de clientes en fichas. Cada ficha podría contener un nombre de cliente, su dirección, ciudad, provincia, código postal y número de teléfono. Cada uno de estos elementos representa una columna posible de una tabla.

Cuando prepare esta lista, no se preocupe si no es perfecta al principio. Simplemente, enumere cada elemento que se le ocurra. Si alguien más va a utilizar la base de datos, pídale también su opinión. Más tarde podrá ajustar la lista.

A continuación, considere los tipos de informes o la correspondencia que desea producir con la base de datos. Por ejemplo, tal vez desee crear un informe de ventas de productos que contenga las ventas por región, o un informe de resumen de inventario con los niveles de inventario de los productos. Es posible que también desee generar cartas modelo para enviárselas a los clientes con un anuncio de una actividad de ventas o una oferta. Diseñe el informe en su imaginación y piense cómo le gustaría que fuera. ¿Qué información incluiría en el informe? Cree un listado de cada elemento. Haga lo mismo para la carta modelo y para cualquier otro informe que tenga pensado crear.



Detenerse a pensar en los informes y en la correspondencia que desea crear le ayudará a identificar los elementos que necesita incluir en la base de datos. Suponga, por ejemplo, que ofrece a sus clientes la oportunidad de inscribirse o borrarse de las actualizaciones periódicas de correo electrónico y desea imprimir un listado de los que han decidido inscribirse. Para registrar esa información, agrega una columna "Enviar correo electrónico" a la tabla de clientes. Para cada cliente, puede definir el campo en Sí o No.

La necesidad de enviar mensajes de correo electrónico a los clientes implica la inclusión de otro elemento. Cuando sepa que un cliente desea recibir mensajes de correo electrónico, tendrá que conocer también la dirección de correo electrónico a la que éstos deben enviarse. Por tanto, tendrá que registrar una dirección de correo electrónico para cada cliente.

Parece lógico crear un prototipo de cada informe o listado de salida y considerar qué elementos necesita para crear el informe. Por ejemplo, cuando examine una carta modelo, puede que se le ocurran algunas ideas. Si desea incluir un saludo (por ejemplo, las abreviaturas "Sr." o "Sra." con las que comienza un saludo), tendrá que crear un elemento de saludo. Además, tal vez desee comenzar las cartas con el saludo "Estimado Sr. García", en lugar de "Estimado Sr. Miguel Ángel García". Esto implicaría almacenar el apellido independientemente del nombre.

Un punto clave que hay que recordar es que debe descomponer cada pieza de información en sus partes lógicas más pequeñas. En el caso de un nombre, para poder utilizar el apellido, dividirá el nombre en dos partes: el nombre y el apellido. Para ordenar un informe por nombre, por ejemplo, sería útil que el apellido de los clientes estuviera almacenado de forma independiente. En general, si desea ordenar, buscar, calcular o generar informes a partir de un elemento de información, debe incluir ese elemento en su propio campo.

Piense en las preguntas que le gustaría que la base de datos contestara. Por ejemplo, ¿cuántas ventas de un determinado producto se cerraron el pasado mes? ¿Dónde viven sus mejores clientes? ¿Quién es el proveedor del producto mejor vendido? Prever esas preguntas le ayudará a determinar los elementos adicionales que necesita registrar.

Una vez reunida esta información, ya puede continuar con el paso siguiente.

[📄 Volver al principio](#)

Dividir la información en tablas

Para dividir la información en tablas, elija las entidades o los temas principales. Por ejemplo, después de buscar y organizar la información de una base de datos de ventas de productos, la lista preliminar podría ser similar a la siguiente:

Cientes <ul style="list-style-type: none">NombreDirecciónCiudad, Estado, CPEnviar correo electrónicoSaludoCorreo electrónico	Productos <ul style="list-style-type: none">Nombre del productoPrecioUnidades almacen...Unidades pedidas
Proveedores <ul style="list-style-type: none">Nombre de la organizaciónNombre del contactoDirecciónCiudad, Estado, CP	Pedidos <ul style="list-style-type: none">Número de pedidoVendedorFecha de pedidoProductoCantidadPrecioTotal

Las entidades principales mostradas aquí son los productos, los proveedores, los clientes y los pedidos. Por tanto, parece lógico empezar con estas cuatro tablas: una para los datos sobre los productos, otra para los datos sobre los proveedores, otra para los datos sobre los clientes y otra para los datos sobre los pedidos. Aunque esto no complete la lista, es un buen punto de partida. Puede seguir ajustando la lista hasta obtener un diseño correcto.

Cuando examine por primera vez la lista preliminar de elementos, podría estar tentado a incluirlos todos ellos en una sola tabla en lugar de en las cuatro tablas mostradas en la ilustración anterior. A continuación le

explicaremos por qué eso no es una buena idea. Considere por un momento la tabla que se muestra a continuación:

Productos y proveedores		
NombreProducto	Proveedores	Dirección
Chai	Exotic Liquids	49 Gilber St.
Chang	Exotic Liquids	49 Gilber St.
Sirope de anís	Exotic Liquids	49 Gilber St.
Espicias Cajun del chef ...	New Orleans Cajun Deli...	P.O. Box 78934

En este caso, cada fila contiene información sobre el producto y su proveedor. Como hay muchos productos del mismo proveedor, la información del nombre y la dirección del proveedor debe repetirse muchas veces, con lo que se malgasta el espacio en disco. Registrar la información del proveedor una sola vez en una tabla Proveedores distinta y luego vincular esa tabla a la tabla Productos es una solución mucho mejor.

Otro problema de este diseño surge cuando es necesario modificar la información del proveedor. Suponga, por ejemplo, que necesita cambiar la dirección de un proveedor. Como ésta aparece en muchos lugares, podría sin querer cambiar la dirección en un lugar y olvidarse de cambiarla en los demás lugares. Ese problema se resuelve registrando la información del proveedor en un único lugar.

Cuando diseñe la base de datos, intente registrar siempre cada dato una sola vez. Si descubre que está repitiendo la misma información en varios lugares, como la dirección de un determinado proveedor, coloque esa información en una tabla distinta.

Por último, suponga que el proveedor Bodega Sol sólo suministra un producto y desea eliminar ese producto pero conservar el nombre del proveedor y la información de dirección. ¿Cómo eliminaría el producto sin perder la información del proveedor? No puede. Como cada registro contiene datos sobre un producto, además de datos sobre un proveedor, no puede eliminar unos sin eliminar los otros. Para mantener estos datos separados, debe dividir la tabla en dos: una tabla para la información sobre los productos y otra tabla para la información sobre los proveedores. Al eliminar un registro de producto sólo se eliminarían los datos del producto y no los datos del proveedor.

Una vez seleccionado el tema representado por una tabla, las columnas de esa tabla deben almacenar datos únicamente sobre ese tema. Por ejemplo, la tabla de productos sólo debe contener datos de productos. Como la dirección del proveedor es un dato del proveedor, pertenece a la tabla de proveedores.

[Volver al principio](#)

Convertir los elementos de información en columnas

Para determinar las columnas de una tabla, decida qué información necesita registrar sobre el tema representado por la tabla. Por ejemplo, para la tabla Clientes, una buena lista de columnas iniciales sería Nombre, Dirección, Ciudad-Provincia-Código postal, Enviar correo electrónico, Saludo y Correo electrónico. Cada registro de la tabla contiene el mismo número de columnas, por lo que puede almacenar información sobre el nombre, dirección, ciudad-provincia-código postal, envío de correo electrónico, saludo y dirección de correo electrónico para cada registro. Por ejemplo, la columna de dirección podría contener las direcciones de los clientes. Cada registro contendrá datos sobre un cliente y el campo de dirección, la dirección de ese cliente.

Cuando haya determinado el conjunto inicial de columnas para cada tabla, puede ajustar con mayor precisión las columnas. Por ejemplo, tiene sentido almacenar los nombres de los clientes en dos columnas distintas (el nombre y el apellido) para poder ordenar, buscar e indizar por esas columnas. De igual forma, la dirección consta en realidad de cinco componentes distintos: dirección, ciudad, provincia, código postal y país o región, y parece lógico también almacenarlos en columnas distintas. Si desea realizar, por ejemplo, una búsqueda o una operación de ordenación o filtrado por provincia, necesita que la información de provincia esté almacenada en una columna distinta.

Debe considerar también si la base de datos va a contener información sólo de procedencia nacional o internacional. Por ejemplo, si piensa almacenar direcciones internacionales, es preferible tener una columna Región en lugar de Provincia, ya que esa columna puede incluir provincias del propio país y regiones de otros países o regiones. De igual forma, es más lógico incluir una columna Región en lugar de Comunidad Autónoma si va a almacenar direcciones internacionales.

En la lista siguiente se incluyen algunas sugerencias para determinar las columnas de la base de datos.

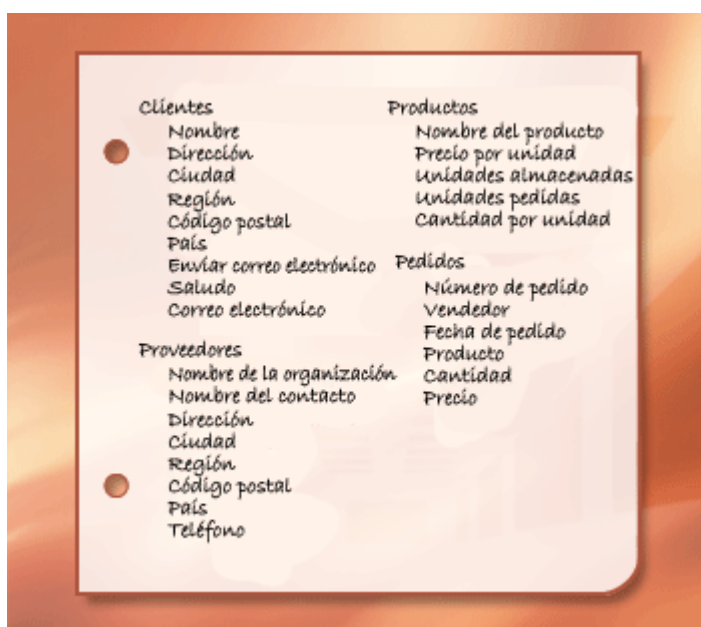
⬇ **No incluya datos calculados**

En la mayoría de los casos, no debe almacenar el resultado de los cálculos en las tablas. En lugar de ello, puede dejar que Access realice los cálculos cuando desee ver el resultado. Suponga, por ejemplo, que tiene un informe Productos bajo pedido que contiene el subtotal de unidades de un pedido para cada categoría de producto de la base de datos. Sin embargo, no hay ninguna tabla que contenga una columna de subtotal Unidades en pedido. La tabla Productos contiene una columna Unidades en pedido que almacena las unidades incluidas en un pedido de cada producto. Con esos

datos, Access calcula el subtotal cada vez que se imprime el informe, pero el subtotal propiamente dicho no debe almacenarse en una tabla.

↓ Almacene la información en sus partes lógicas más pequeñas

Puede ceder a la tentación de habilitar un único campo para los nombres completos o para los nombres de productos junto con sus descripciones. Si combina varios tipos de información en un campo, será difícil recuperar datos individuales más adelante. Intente dividir la información en partes lógicas. Por ejemplo, cree campos distintos para el nombre y el apellido, o para el nombre del producto, la categoría y la descripción.



Clientes	Productos
Nombre	Nombre del producto
Dirección	Precio por unidad
Ciudad	Unidades almacenadas
Región	Unidades pedidas
Código postal	Cantidad por unidad
País	
Enviar correo electrónico	Pedidos
Saludo	Número de pedido
Correo electrónico	Vendedor
	Fecha de pedido
Proveedores	Producto
Nombre de la organización	Cantidad
Nombre del contacto	Precio
Dirección	
Ciudad	
Región	
Código postal	
País	
Teléfono	

Una vez ajustadas las columnas de datos de cada tabla, ya puede seleccionar la clave principal de cada tabla.

[📄 Volver al principio](#)

Especificar claves principales

Cada tabla debe incluir una columna o conjunto de columnas que identifiquen inequívocamente cada fila almacenada en la tabla. Ésta suele ser un número de identificación exclusivo, como un número de identificador de empleado o un número de serie. En la terminología de bases de datos, esta información recibe el nombre de **clave principal** de la tabla. Access utiliza los campos de clave principal para asociar rápidamente datos de varias tablas y reunir automáticamente esos datos.

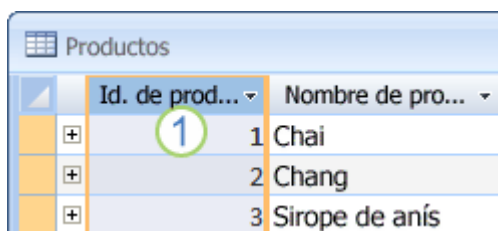
Si ya tiene un identificador exclusivo para una tabla, como un número de producto que identifica inequívocamente cada producto del catálogo, puede utilizar ese identificador como clave principal de la tabla, pero sólo si los valores de esa columna son siempre diferentes para cada registro. No puede tener valores duplicados en una clave principal. Por ejemplo, no utilice los nombres de las personas como clave principal, ya que los nombres no son exclusivos. Es muy fácil que dos personas tengan el mismo nombre en la misma tabla.

Una clave principal siempre debe tener un valor. Si el valor de una columna puede quedar sin asignar o vacío (porque no se conoce) en algún momento, no puede utilizarlo como componente de una clave principal.

Debe elegir siempre una clave principal cuyo valor no cambie. En una base de datos con varias tablas, la clave principal de una tabla se puede utilizar como referencia en las demás tablas. Si la clave principal cambia, el cambio debe aplicarse también a todos los lugares donde se haga referencia a la clave. Usar una clave principal que no cambie reduce la posibilidad de que se pierda su sincronización con las otras tablas en las que se hace referencia a ella.

A menudo, se utiliza como clave principal un número único arbitrario. Por ejemplo, puede asignar a cada pedido un número de pedido distinto. La única finalidad de este número de pedido es identificar el pedido. Una vez asignado, nunca cambia.

Si piensa que no hay ninguna columna o conjunto de columnas que pueda constituir una buena clave principal, considere la posibilidad de utilizar una columna que tenga el tipo de datos Autonumérico. Cuando se utiliza el tipo de datos Autonumérico, Access asigna automáticamente un valor. Este tipo de identificador no es "fáctico", es decir, no contiene información objetiva sobre la fila que representa. Los identificadores de este tipo son perfectos para usarlos como claves principales, ya que no cambian. Una clave principal que contiene datos sobre una fila, como un número de teléfono o el nombre de un cliente, es más probable que cambie, ya que la propia información "fáctica" podría cambiar.

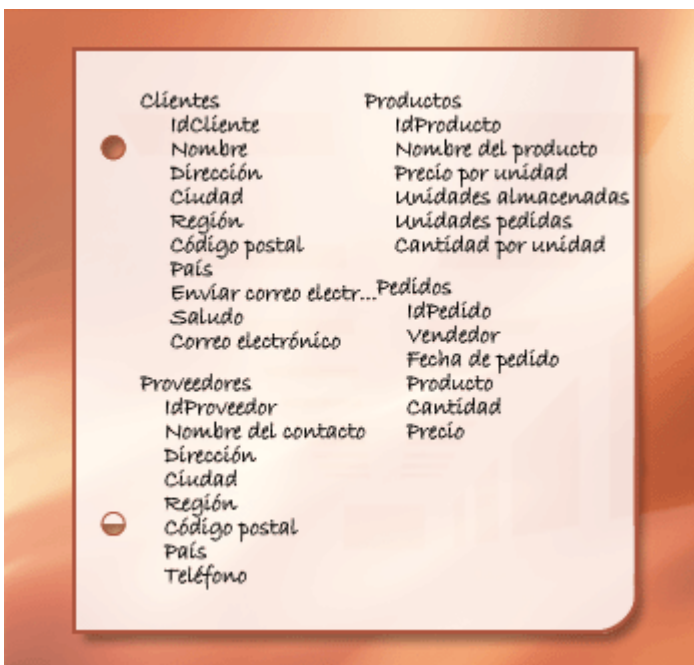


	Id. de prod...	Nombre de pro...
	1	Chai
	2	Chang
	3	Sirope de anís

1 Una columna establecida en el tipo de datos Autonumérico suele constituir una buena clave principal. No hay dos identificadores de producto iguales.

En algunos casos, tal vez considere conveniente utilizar dos o más campos juntos como clave principal de una tabla. Por ejemplo, una tabla Detalles de pedidos que contenga artículos de línea de pedidos tendría dos columnas en su clave principal: Id. de pedido e Id. de producto. Cuando una clave principal está formada por más de una columna se denomina clave compuesta.

Para la base de datos de ventas de productos, puede crear una columna autonumérica para cada una de las tablas que funcione como clave principal: IdProducto para la tabla Productos, IdPedido para la tabla Pedidos, IdCliente para la tabla Clientes e IdProveedores para la tabla Proveedores.



Clientes	Productos
IdCliente	IdProducto
Nombre	Nombre del producto
Dirección	Precio por unidad
Ciudad	Unidades almacenadas
Región	Unidades pedidas
Código postal	Cantidad por unidad
País	
Enviar correo electr...	Pedidos
Saludo	IdPedido
Correo electrónico	Vendedor
	Fecha de pedido
	Producto
	Cantidad
	Precio
Proveedores	
IdProveedor	
Nombre del contacto	
Dirección	
Ciudad	
Región	
Código postal	
País	
Teléfono	

[Volver al principio](#)

Crear relaciones entre las tablas

Ahora que ha dividido la información en tablas necesita un modo de reunir de nuevo la información de forma provechosa. Por ejemplo, el siguiente formulario incluye información de varias tablas.

Pedidos

Factura a: Alfreds Futterkiste
Obere Str. 57
Berlin 12209
Alemania

Enviar a: Alfreds Futterkiste
Obere Str. 57
Berlin 12209
Alemania

Vendedor: Suyama, Michael

Enviar por: Rápido Unido Federal

Id. de pe...: 10643 **Fecha:** 25.08.1997 **Fecha adecuada:** 22.08.1997

Productos	Precio un...	Cantidad	Descue...	Precio total
Spegesild	12,00 \$	2	25%	18,00
Chartreuse verte	18,00 \$	21	25%	283,50
Rossle Sauerkraut	45,60 \$	15	25%	513,00
*			0%	

Subtotal: 814,50 \$
Carga: 29,46 \$
Total: 843,96 \$

[Imprimir factura](#)

1 La información de este formulario procede de la tabla Clientes...

2 ...la tabla Empleados...

3 ...la tabla Pedidos...

4 ...la tabla Productos...

5 ...y la tabla Detalles de pedidos.

Access es un sistema de administración de bases de datos relacionales. En una base de datos relacional, la información se divide en tablas distintas en función del tema. A continuación, se utilizan relaciones entre las tablas para reunir la información según se precise.

[Volver al principio](#)

Crear una relación de uno a varios

Considere este ejemplo: las tablas Proveedores y Productos de la base de datos de pedidos de productos. Un proveedor puede suministrar cualquier número de productos y, por consiguiente, para cada proveedor representado en la tabla Proveedores, puede haber muchos productos representados en la tabla Productos. La relación entre la tabla Proveedores y la tabla Productos es, por tanto, una relación de uno a varios.

The screenshot shows two data tables from Microsoft Access. The top table, 'Proveedores', has columns 'Id. de proveedor' and 'Nombre de la compañía'. The bottom table, 'Productos', has columns 'Id. de producto', 'Nombre de producto', and 'Proveedor'. A red arrow points from the 'Id. de proveedor' field in the 'Proveedores' table to the 'Proveedor' field in the 'Productos' table, indicating a one-to-many relationship.

Id. de proveedor	Nombre de la compañía
1	Exotic Liquids

Id. de producto	Nombre de producto	Proveedor
1	Chai	1
2	Chang	1
3	Sirope de anís	1

Para representar una relación de uno a varios en el diseño de la base de datos, tome la clave principal del lado "uno" de la relación y agréguela como columna o columnas adicionales a la tabla en el lado "varios" de la relación. En este caso, por ejemplo, agregaría la columna Id. de proveedor de la tabla Proveedores a la tabla Productos. Access utilizaría entonces el número de identificador de proveedor de la tabla Productos para localizar el proveedor correcto de cada producto.

La columna Id. de proveedor de la tabla Productos se denomina clave externa. Una clave externa es la clave principal de otra tabla. La columna Id. de proveedor de la tabla Productos es una clave externa porque también es la clave principal en la tabla Proveedores.

Cientes	Productos
IdCliente	IdProducto
Nombre	Nombre del producto
Dirección	Precio por unidad
Ciudad	Unidades almacen...
Región	Unidades pedidas
Código postal	Cantidad por unidad
País	IdProveedor
Enviar correo electrónico	
Saludo	Pedidos
Correo electrónico	IdPedido
	Vendedor
	Fecha de pedido
Proveedores	Producto
IdProveedor	Cantidad
Nombre del contacto	Precio
Dirección	
Ciudad	
Región	
Código postal	
País	
Teléfono	

El punto de partida para la unión de tablas relacionadas se proporciona estableciendo parejas de claves principales y claves externas. Si no está seguro de las tablas que deben compartir una columna común, al identificar una relación de uno a varios se asegurará de que las dos tablas implicadas requerirán una columna compartida.

[Volver al principio](#)

Crear una relación de varios a varios

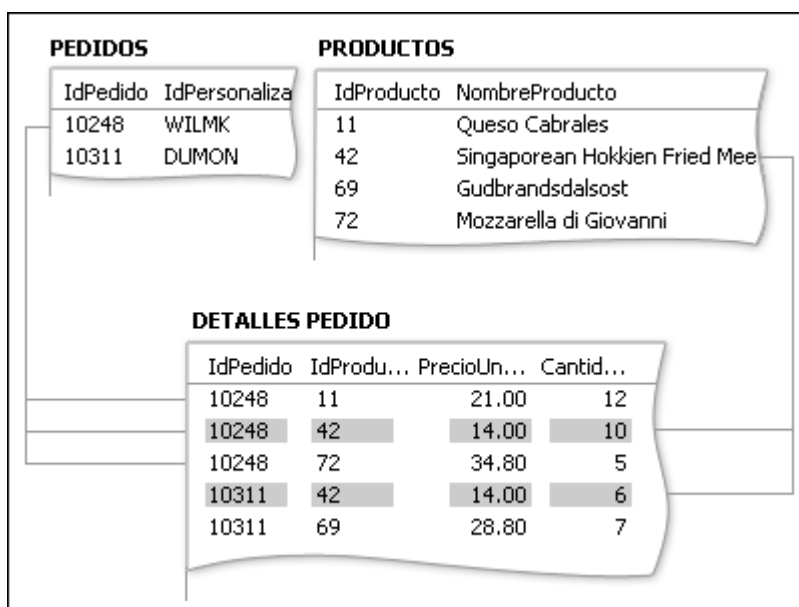
Considere la relación entre la tabla Productos y la tabla Pedidos.

Un solo pedido puede incluir varios productos. Por otro lado, un único producto puede aparecer en muchos pedidos. Por tanto, para cada registro de la tabla Pedidos puede haber varios registros en la tabla Productos. Y para cada registro de la tabla Productos puede haber varios registros en la tabla Pedidos. Este tipo de relación se denomina relación de varios a varios porque para un producto puede haber varios pedidos, y para un pedido puede haber varios productos. Tenga en cuenta que para detectar las relaciones de varios a varios entre las tablas, es importante que considere ambas partes de la relación.

Los temas de las dos tablas (pedidos y productos) tienen una relación de varios a varios. Esto presenta un problema. Para comprender el problema, imagine qué sucedería si intenta crear la relación entre las dos tablas agregando el campo Id. de producto a la tabla Pedidos. Para que haya más de un producto por pedido,

necesita más de un registro en la tabla Pedidos para cada pedido y, en ese caso, tendría que repetir la información de pedido para cada fila relacionada con un único pedido, lo que daría lugar a un diseño ineficaz que podría producir datos inexactos. El mismo problema aparece si coloca el campo Id. de pedido en la tabla Productos: tendría varios registros en la tabla Productos para cada producto. ¿Cómo se soluciona este problema?

La solución a este problema consiste en crear una tercera tabla que descomponga la relación de varios a varios en dos relaciones de uno a varios. Insertaría la clave principal de cada una de las dos tablas en la tercera tabla y, por consiguiente, la tercera tabla registraría todas las apariciones o instancias de la relación.



Cada registro de la tabla Detalles de pedidos representa un artículo de línea de un pedido. La clave principal de la tabla Detalles de pedidos consta de dos campos: las claves externas de las tablas Pedidos y Productos. El campo Id. de pedido no se puede utilizar en solitario como clave principal, ya que un pedido puede tener varios artículos de línea. El identificador de pedido se repite para cada artículo de línea del pedido, por lo que el campo no contiene valores únicos. Tampoco serviría utilizar solamente el campo Id. de producto, porque un producto puede aparecer en varios pedidos. Pero los dos campos juntos producen un valor exclusivo para cada registro.

En la base de datos de ventas de productos, la tabla Pedidos y la tabla Productos no se relacionan directamente entre sí, sino indirectamente a través de la tabla Detalles de pedidos. La relación de varios a varios entre los pedidos y los productos se representa en la base de datos mediante dos relaciones de uno a varios:

- ↓ La tabla Pedidos y la tabla Detalles de pedidos tienen una relación de uno a varios. Cada pedido tiene varios artículos de línea, pero cada artículo está asociado a un único pedido.
- ↓ La tabla Productos y la tabla Detalles de pedidos tienen una relación de uno a varios. Cada producto puede tener varios artículos asociados, pero cada artículo de línea hace referencia únicamente a un producto.

Desde la tabla Detalles de pedidos puede determinar todos los productos de un determinado pedido, así como todos los pedidos de un determinado producto.

Después de incorporar la tabla Detalles de pedidos, la lista de tablas y campos sería similar a la siguiente:

Clientes	Productos
IdCliente	IdProducto
Nombre	Nombre del producto
Dirección	Precio por unidad
Ciudad	Unidades almacen...
Región	Unidades pedidas
Código postal	Cantidad por unidad
País	IdProveedor
Enviar correo electrónico	Pedidos
Saludo	IdPedido
Correo electrónico	Vendedor
Proveedores	Fecha de pedido
IdProveedor	IdCliente
Nombre de la organiz...	NombreEnvío
Nombre del contacto	DirecciónEnvío
Dirección	CiudadDestino
Ciudad	RegiónDestino
Región	CódigoPostalEnvío
Código postal	PaísEnvío
País	Detalles pedido
Teléfono	IdPedido
	IdProducto
	PrecioUnidad
	Cantidad

[🏠 Volver al principio](#)

Crear una relación de uno a uno


Otro tipo de relación es la relación de uno a uno. Suponga, por ejemplo, que necesita registrar información complementaria sobre productos que apenas va a necesitar o que sólo se aplica a unos pocos productos.

Como no necesita la información con frecuencia, y como almacenar la información en la tabla Productos crearía un espacio vacío para todos los productos que no necesitan esa información, la coloca en una tabla distinta. Al igual que en la tabla Productos, utiliza el identificador de producto como clave principal. La relación entre esta tabla complementaria y la tabla Productos es una relación de uno a uno. Para cada registro de la tabla Productos hay un único registro coincidente en la tabla complementaria. Cuando identifique esta relación, ambas tablas deben compartir un campo común.

Cuando necesite crear una relación de uno a uno en la base de datos, considere si puede incluir la información de las dos tablas en una tabla. Si no desea hacer eso por algún motivo (quizás porque se crearía una gran cantidad de espacio vacío), puede representar esa relación en su diseño guiándose por las pautas siguientes:

- ↓ Si las dos tablas tienen el mismo tema, probablemente podrá definir la relación utilizando la misma clave principal en ambas tablas.
- ↓ Si las dos tablas tienen temas diferentes con claves principales distintas, elija una de las tablas (cualquiera de ellas) e inserte su clave principal en la otra tabla como clave externa.

Determinar las relaciones entre las tablas le ayudará a asegurarse de que tiene las tablas y columnas correctas. Cuando existe una relación de uno a uno o de uno a varios, las tablas implicadas deben compartir una o varias columnas comunes. Cuando la relación es de varios a varios, se necesita una tercera tabla para representar la relación.

 [Volver al principio](#)

Ajustar el diseño

Cuando tenga las tablas, los campos y las relaciones necesarias, debe crear y rellenar las tablas con datos de ejemplo y probar que funcionan con la información: creando consultas, agregando nuevos registros, etc. Esto le permitirá encontrar posibles problemas, como la necesidad de agregar una columna que olvidó insertar durante la fase de diseño, o dividir una tabla en dos tablas para eliminar datos duplicados.

Compruebe si puede usar la base de datos para obtener las respuestas que desea. Cree formularios e informes provisionales y compruebe si muestran los datos según lo previsto. Compruebe si existen datos duplicados innecesarios y, si encuentra alguno, modifique el diseño para eliminar la duplicación.

Cuando pruebe la base de datos inicial, probablemente se dará cuenta de que se puede mejorar. Éstas son algunas comprobaciones que puede hacer:

- ↓ ¿Olvidó incluir alguna columna? Y, en ese caso, ¿pertenece la información a alguna de las tablas existentes? Si se trata de información sobre otro tema, tal vez necesite crear otra tabla. Cree una columna para cada elemento de información que desee registrar. Si la información no se puede calcular a partir de otras columnas, es probable que necesite una nueva columna para esa información.
- ↓ ¿Hay alguna columna innecesaria porque se puede calcular con los campos existentes? Si un elemento de información se puede calcular a partir de otras columnas existentes (como un descuento calculado a partir del precio de venta al público), normalmente es preferible que se calcule en lugar de crear una nueva columna.
- ↓ ¿Ha proporcionado información duplicada en alguna de las tablas? Si es así, probablemente tendrá que dividir la tabla en dos tablas que tengan una relación de uno a varios.
- ↓ ¿Tiene tablas con muchos campos, un número limitado de registros y muchos campos vacíos en cada registro? En ese caso, considere la posibilidad de volver a diseñar la tabla de forma que tenga menos campos y más registros.
- ↓ ¿Ha dividido cada elemento de información en sus partes lógicas más pequeñas? Si necesita generar informes, ordenar, buscar o calcular a partir de un elemento de información, incluya ese elemento en su propia columna.
- ↓ ¿Contiene cada columna datos sobre el tema de la tabla? Si una columna no contiene información sobre el tema de la tabla, pertenece a una tabla distinta.
- ↓ ¿Están representadas todas las relaciones entre las tablas mediante campos comunes o mediante una tercera tabla? Las relaciones de uno a uno y de uno a varios requieren columnas comunes. Las relaciones de varios a varios requieren una tercera tabla.

Ajustar la tabla Productos

Suponga que cada producto de la base de datos de ventas de productos pertenece a una categoría general, como bebidas, condimentos o marisco. La tabla Productos podría incluir un campo que mostrara la categoría de cada producto.

Suponga que después de examinar y ajustar el diseño de la base de datos, decide almacenar una descripción de la categoría junto con su nombre. Si agrega un campo Descripción de categoría a la tabla Productos, tendría que repetir la descripción de cada categoría para cada producto perteneciente a dicha categoría, lo cual no es una buena solución.

Una solución mejor es convertir las categorías en un nuevo tema de la base de datos, con su propia tabla y su propia clave principal. A continuación, puede agregar la clave principal de la tabla Categorías a la tabla Productos como clave externa.

Las tablas Categorías y Productos tienen una relación de uno a varios: una categoría puede incluir varios productos, pero un producto pertenece únicamente a una categoría.

Cuando examine las estructuras de las tablas, compruebe si existen grupos extensibles. Por ejemplo, considere una tabla con las siguientes columnas:

↓ Id. de producto

↓ Nombre

↓ Id1 de producto

↓ Nombre1

↓ Id2 de producto

↓ Nombre2

↓ Id3 de producto

↓ Nombre3

Aquí, cada producto es un grupo extensible de columnas que se diferencia de los demás solamente por el número agregado al final del nombre de columna. Si tiene columnas numeradas de esta forma, debe revisar el diseño.

Un diseño como éste tiene varios defectos. Para empezar, obliga a crear un límite máximo de número de productos. En cuanto supere ese límite, deberá agregar un nuevo grupo de columnas a la estructura de la tabla, lo que supone una tarea administrativa importante.

Otro problema es que se malgastará el espacio en aquellos proveedores que tengan menos que el número máximo de productos, ya que las columnas adicionales quedarán en blanco. El defecto más grave de este diseño es que muchas tareas son difíciles de realizar, como ordenar o indizar la tabla por identificador de producto o nombre.

Siempre que aparezcan grupos extensibles, revise atentamente el diseño con vistas a dividir la tabla en dos. En el ejemplo anterior, sería conveniente usar dos tablas, una para proveedores y otra para productos, vinculadas por el identificador de proveedor.

 [Volver al principio](#)

Aplicar las reglas de normalización

En el siguiente paso del diseño, puede aplicar las reglas de normalización de datos (denominadas a veces simplemente reglas de normalización). Estas reglas sirven para comprobar si las tablas están estructuradas correctamente. El proceso de aplicar las reglas al diseño de la base de datos se denomina normalizar la base de datos o, simplemente, normalización.

La normalización es más útil una vez representados todos los elementos de información y después de haber definido un diseño preliminar. La idea es asegurarse de que se han dividido los elementos de información en las tablas adecuadas. Lo que la normalización no puede hacer es garantizar que se dispone de los elementos de datos correctos para empezar a trabajar.

Las reglas se aplican consecutivamente en cada paso para garantizar que el diseño adopta lo que se conoce como "forma normal". Hay cinco formas normales ampliamente aceptadas: de la primera forma normal a la quinta forma normal. En este artículo se abordan las tres primeras, porque todas ellas son necesarias para la mayoría de los diseños de base de datos.

Primera forma normal

La primera forma normal establece que en cada intersección de fila y columna de la tabla existe un valor y nunca una lista de valores. Por ejemplo, no puede haber un campo denominado Precio en el que se incluya más de un precio. Si considera cada intersección de filas y columnas como una celda, cada celda sólo puede contener un valor.

Segunda forma normal

La segunda forma normal exige que cada columna que no sea clave dependa por completo de toda la clave principal y no sólo de parte de la clave. Esta regla se aplica cuando existe una clave principal formada por varias columnas. Suponga, por ejemplo, que existe una tabla con las siguientes columnas, de las cuales Id. de pedido e Id. de producto forman la clave principal:

↓ Id. de pedido (clave principal)

↓ Id. de producto (clave principal)

↓ Nombre de producto

Este diseño infringe los requisitos de la segunda forma normal, porque Nombre de producto depende de Id. de producto, pero no de Id. de pedido, por lo que no depende de toda la clave principal. Debe quitar Nombre de producto de la tabla, ya que pertenece a una tabla diferente (a la tabla Productos).

Tercera forma normal

La tercera forma normal exige no sólo que cada columna que no sea clave dependa de toda la clave principal, sino también que las columnas que no sean clave sean independientes unas de otras.

O dicho de otra forma: cada columna que no sea clave debe depender de la clave principal y nada más que de la clave principal. Por ejemplo, considere una tabla con las siguientes columnas:

↓ IdProducto (clave principal)

↓ Nombre

↓ PVP

↓ Descuento

Suponga que la columna Descuento depende del precio de venta al público (PVP) sugerido. Esta tabla infringe los requisitos de la tercera forma normal porque una columna que no es clave, la columna Descuento, depende de otra columna que no es clave, la columna PVP. La independencia de las columnas implica que debe poder cambiar cualquier columna que no sea clave sin que ninguna otra columna resulte afectada. Si cambia un valor en el campo PVP, la columna Descuento cambiaría en consecuencia e infringiría esa regla. En este caso, la columna Descuento debe moverse a otra tabla cuya clave sea PVP.

[↑ Volver al principio](#)

Para obtener más información

Para obtener más información sobre los principios básicos de diseño de una tabla, vea el artículo [Crear tablas en una base de datos](#).

Para obtener información adicional sobre el diseño de la base de datos, vea los libros siguientes:

- ↓ Hernandez, Michael J. *Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design, Second Edition*. Addison-Wesley Professional. 2003.
- ↓ Fleming, Candace C. von Halle, Barbara. *Handbook of Relational Database Design*. Addison-Wesley Professional. 1989.
- ↓ Riordan, Rebecca M. *Designing Effective Database Systems*. Addison-Wesley Professional. 2005.