



## Guía de Base

### *Capítulo 10*

### *Mantenimiento de bases de datos*

## Derechos de autor

---

Este documento tiene derechos de autor © 2021 por el equipo de documentación. Los colaboradores se listan más abajo. Se puede distribuir y modificar bajo los términos de la [GNU General Public License](#) versión 3 o posterior o la [Creative Commons Attribution License](#), versión 4.0 o posterior.

Todas las marcas registradas mencionadas en esta guía pertenecen a sus propietarios legítimos.

## Colaboradores

Este libro está adaptado de versiones anteriores del mismo.

### De esta edición

---

Pulkit Krishna  
Juan Peramos

Dan Lewis  
Juan Carlos Sanz Cabrero

Jean Hollis Weber  
Jonatán Perren

### De ediciones previas

Jochen Schiffers  
Hazel Russman

Robert Großkopf

Jost Lange

Mayo de 2020. 2 Comentarios y sugerencias

---

Puede dirigir cualquier clase de comentario o sugerencia acerca de este documento a: [documentation@es.libreoffice.org](mailto:documentation@es.libreoffice.org).



### Nota

Todo lo que envíe a la lista de correo, incluyendo su dirección de correo y cualquier otra información personal que escriba en el mensaje se archiva públicamente y no puede ser borrado.

---

## Fecha de publicación y versión del programa

Versión en español publicada el 17 de junio de 2021. Basada en la versión 6.2 de LibreOffice.

## Uso de LibreOffice en macOS

---

Algunas pulsaciones de teclado y opciones de menú son diferentes en macOS de las usadas en Windows y Linux. La siguiente tabla muestra algunas sustituciones comunes para las instrucciones dadas en este capítulo. Para una lista detallada vea la ayuda de la aplicación.

<b>Windows o Linux</b>	<b>Equivalente en Mac</b>	<b>Efecto</b>
<b>Herramientas &gt; Opciones</b> opción de menú	<b>LibreOffice &gt; Preferencias</b>	Acceso a las opciones de configuración
<i>Clic con el botón derecho</i>	<i>Control+clic o clic derecho</i> depende de la configuración del equipo	Abre menú contextual
<i>Ctrl (Control)</i>	⌘ ( <i>Comando</i> )	Utilizado con otras teclas
<i>F5</i>	<i>Mayúscula+⌘+F5</i>	Abre el navegador
<i>F11</i>	⌘+T	Abre la ventana de estilos y formato

## Contenido

---

<b>Derechos de autor</b> .....	<b>2</b>
Colaboradores.....	2
<b>De esta edición</b> .....	<b>2</b>
De ediciones previas.....	2
Fecha de publicación y versión del programa.....	2
<b>Uso de LibreOffice en macOS</b> .....	<b>2</b>
<b>Observaciones generales sobre el mantenimiento de bases de datos</b> .....	<b>5</b>
<b>Compactar una base de datos</b> .....	<b>5</b>
<b>Restablecer valores automáticos</b> .....	<b>5</b>
<b>Consultar propiedades de la base de datos</b> .....	<b>6</b>
<b>Exportar datos</b> .....	<b>6</b>
<b>Tablas de prueba para entradas innecesarias</b> .....	<b>7</b>
Probar entradas utilizando la definición de relación.....	7
Editar entradas usando formularios y subformularios.....	8
Consultas para encontrar entradas huérfanas.....	9
<b>Velocidad de búsqueda en base de datos</b> .....	<b>10</b>
Efecto de consultas.....	10
Efecto de Listados y Cuadros combinados.....	10
Influencia del sistema de base de datos utilizado.....	10

## Observaciones generales sobre el mantenimiento de bases de datos

---

La alteración frecuente de los datos en una base de datos, en particular muchas eliminaciones, tiene dos efectos. Primero, la base de datos crece de manera constante aunque en realidad no contenga más datos. En segundo lugar, la clave primaria creada automáticamente continúa incrementándose independientemente de si es realmente necesario o no. El mantenimiento importante se describe en este capítulo.

### Compactar una base de datos

---

El comportamiento de HSQLDB es preservar el espacio de almacenamiento para los registros eliminados. Las bases de datos que están llenas de datos de prueba, especialmente si incluyen imágenes, conservan el mismo tamaño incluso si todos estos registros se eliminan posteriormente. Esto se debe a una propiedad de las claves principales de cada tabla. El archivo de la base de datos contiene el último valor utilizado para cada clave primaria. Cuando se crea un nuevo registro dentro de una tabla, se le asigna el siguiente valor.

Para liberar este espacio de almacenamiento, los registros de la base de datos deben reescribirse (tablas, descripciones de tablas, etc.). Esto se puede hacer abriendo cada tabla y eliminando todos sus registros. Se debe tener cuidado al tratar con tablas vinculadas.

Use **Herramientas > Relaciones** para determinar qué tabla debe tener sus datos eliminados. Observe las dos tablas. La que tiene su clave principal como parte de la relación es la tabla cuyos datos deben eliminarse. Cierre el diálogo *Relaciones*. Seleccione el icono *Tablas* en la ventana principal de Base. Luego haga doble clic en la tabla para mostrar los datos. Elimine sus datos. Guarde la tabla y luego la base de datos. Después de hacer esto, se necesita que se escriba en el archivo de documento de la base de datos: para hacer esto, cierre LibreOffice. Esto también compactará los archivos de la base de datos.

Cierre LibreOffice y vuelva a abrirlo si va a usarlo nuevamente.

### Restablecer valores automáticos

---

Se crea una base de datos, todas las funciones posibles se prueban con ejemplos y se realizan correcciones hasta que todo funcione. En este momento, en promedio, muchos valores de la clave primaria habrán aumentado a más de 100. Sería contraproducente si la clave primaria se ha configurado para incrementarse automáticamente, como es común. Si las tablas se vacían en preparación para el uso normal o antes de pasar la base de datos a otra persona, la clave primaria continúa incrementándose desde su posición actual, en lugar de restablecerse a cero.

La siguiente orden SQL, ingresada usando **Herramientas > SQL**, permite restablecer el valor inicial:

```
ALTER TABLE "NombreDeLaTabla" ALTER COLUMN "ID" RESTART WITH New value
```

Esto supone que el campo de la clave primaria tiene por nombre «ID» y se ha definido como un campo de autonumeración. El nuevo valor debe ser el que desea que se cree automáticamente para el próximo nuevo registro. Así, por ejemplo, si los registros actuales suben a 4, el nuevo valor debería ser 5 sin alterar el campo ID. El primer valor de ID será el nuevo valor en la orden SQL anterior.

## Consultar propiedades de la base de datos

---

Toda la información en las tablas de la base de datos se almacena en forma de tabla en una parte separada de HSQLDB. Se puede llegar a esta área separada utilizando el nombre INFORMATION\_SCHEMA.

La siguiente consulta se puede utilizar para encontrar nombres de campos, tipos de campos, tamaños de columna y valores predeterminados. Aquí hay un ejemplo para una tabla llamada Searchtable.

```
SELECT "COLUMN_NAME", "TYPE_NAME", "COLUMN_SIZE", "COLUMN_DEF" AS "Default Value" FROM "INFORMATION_SCHEMA"."SYSTEM_COLUMNS" WHERE "TABLE_NAME" = 'Searchtable' ORDER BY "ORDINAL_POSITION"
```

Todas las tablas especiales en HSQLDB se describen en el Apéndice A de este libro. La información sobre el contenido de estas tablas se obtiene más fácilmente mediante consultas directas:

```
SELECT * FROM "INFORMATION_SCHEMA"."SYSTEM_PRIMARYKEYS"
```

El asterisco garantiza que se muestren todas las columnas disponibles de la tabla. La tabla buscada arriba proporciona información esencial sobre las claves principales de las distintas tablas.

Esta información es útil sobre todo para macros. En lugar de tener que proporcionar información detallada sobre cada tabla o base de datos recién creada, los procedimientos se escriben para obtener esta información directamente de la base de datos y, por lo tanto, son de aplicación universal. La base de datos de ejemplo muestra esto, entre otras cosas, en uno de los módulos de mantenimiento, donde se determinan las claves externas.

## Exportar datos

---

A la par con la posibilidad de exportar datos abriendo el archivo \*.odb, hay un método mucho más simple. Directamente en la interfaz Base, puede usar **Herramientas > SQL** para ingresar una orden simple que, en las bases de datos del servidor, está reservado para el administrador del sistema.

```
SCRIPT 'database name'
```

Esto crea una extracción SQL completa de la base de datos con todas las definiciones de tabla, relaciones entre tablas y registros. No se accede a las consultas y formularios, ya que se crearon en la interfaz de usuario y no se almacenan en la base de datos interna. Sin embargo, todas las vistas están incluidas.



### Nota

Este procedimiento se puede usar para actualizar una base de datos integrada para conectarse a la base de datos con HSQLDB 2.50. Nuevamente, las consultas y los formularios tienen que ser reemplazados.

Por defecto, el archivo resultante es un archivo de texto normal. También se puede exportar en binario o comprimido (en formato comprimido), especialmente para grandes bases de datos. Sin embargo, esto hace que volver a importarlo en LibreOffice sea algo más complicado.

El formato del archivo exportado se puede cambiar usando:

```
SET SCRIPTFORMAT {TEXT | BINARY | COMPRESSED};
```

Para exportar el archivo es necesario usar este código SQL línea por línea:

```
SCRIPT 'database name';
```

```
SET SCRIPTFORMAT {TEXT | BINARY | COMPRESSED};
SHUTDOWN SCRIPT;
CHECKPOINT;
```

Esto exporta el archivo de texto *NombreDeLaBaseDeDatos* en la carpeta de inicio con la información de la base de datos.

El archivo se puede leer usando **Herramientas > SQL**, creando una nueva base de datos con los mismos datos. En el caso de una base de datos interna, las siguientes líneas deben eliminarse antes de la importación:

```
CREATE SCHEMA PUBLIC AUTHORIZATION DBA
CREATE USER SA PASSWORD ""
GRANT DBA TO SA
SET WRITE_DELAY 60
SET SCHEMA PUBLIC
```

Estas entradas tratan con el perfil de usuario y otras configuraciones predeterminadas, que ya están configuradas para las bases de datos internas de LibreOffice. Como resultado, aparece un mensaje de error si alguna de estas líneas está presente. Estas se encuentran directamente antes del contenido que se insertará en las tablas con la orden INSERT.

Para importar este archivo, su contenido debe dividirse en múltiples archivos de texto creados por un simple programa de edición de texto. El primer archivo debe contener todas las tablas y vistas de creación. Copie todas las líneas de la primera línea que comienza con CREATE TABLE en la línea que está arriba de la línea que contiene INSERT INTO. Pegue esto en el primer archivo. Luego, copie y pegue el resto del archivo en el segundo archivo.

Hay un límite para el tamaño del segundo archivo: debe ser inferior a 65 kB. Si es más grande, también debe dividirse en archivos de texto más pequeños cortando y pegando. Solo asegúrese de que la línea superior de cada uno de estos archivos nuevos comience con INSERT INTO. Una forma de hacer esto es cortar de abajo hacia arriba a esa línea.

## Tablas de prueba para entradas innecesarias

Una base de datos consta de una o más tablas principales, que contienen claves foráneas de otras tablas. En la base de datos de ejemplo, estas son las tablas *Media* y *Address*. En la tabla *Address*, la clave principal del código postal aparece como una clave externa. Si una persona se muda a un nuevo hogar, la dirección cambia. El resultado puede ser que ya no exista una clave externa *Postcode\_ID* correspondiente a este código postal. Por lo tanto, en principio, el código postal en sí podría eliminarse. Sin embargo, no es común durante el uso normal que el registro ya no sea necesario. Hay varias formas de evitar que surja este tipo de problema.

## Probar entradas utilizando la definición de relación

La integridad de los datos se puede garantizar al definir las relaciones. En otras palabras, puede evitar que la eliminación o alteración de las claves provoque errores en la base de datos. Se puede acceder al siguiente diálogo a través de **Herramientas > Relaciones**, seguido de un clic derecho en el conector entre dos tablas (Ver Figura 81).

Aquí se consideran las tablas *Address* y *Street*. Todas las acciones especificadas se aplican a la tabla *Address*, que contiene la clave externa *Street\_ID*. Las opciones de actualización se refieren a una actualización del campo *ID* en la tabla *Street*. Si la clave numérica en el campo "Street"."ID" es alterada, la opción *Ninguna acción* indica que la base de datos resiste este cambio si aparece un "Street"."ID" con ese número de clave como clave foránea en la tabla *Address*.

*Actualización en cascada* indica que el número clave simplemente se hereda. Si la calle «Burgring» en la tabla *Street* tiene la ID «3» y también está representada en

"Address"."Street\_ID", la ID se puede modificar de forma segura, por ejemplo a «67», los valores correspondientes de "Address"."Street\_ID" se cambiarán automáticamente a «67».



Figura 1

Si se elige *Definir NULL*, la modificación de la ID hace que "Address"."Street\_ID" sea un campo vacío.

Las opciones de eliminación a la derecha funcionan de manera similar.

Para ambas opciones, la posibilidad de establecer valores predeterminados no está permitida en la GUI actualmente, ya que las configuraciones predeterminadas de la GUI son diferentes de las de la base de datos. Ver *Capítulo 3, Tablas*.

La definición de relaciones ayuda a mantener las relaciones limpias, pero no elimina los registros innecesarios que proporcionan su clave principal como clave externa en la relación. Para el ejemplo anterior, puede haber cualquier número de calles sin las direcciones correspondientes.

## Editar entradas usando formularios y subformularios

En principio, toda la interrelación entre tablas se puede mostrar dentro de los formularios. Esto es más fácil, por supuesto, cuando una tabla está relacionada con solo otra tabla. Por lo tanto, en el siguiente ejemplo, la clave principal de *Author* se convierte en la clave externa en la tabla *rel\_Media\_Author*. La tabla *rel\_Media\_Author* también contiene una clave foránea de *Media*, de modo que la siguiente disposición muestra una relación n:m con tres formularios. Cada una se presenta a través de una tabla.

La primera figura muestra que el libro *I hear you knocking* pertenece al autor Dave Edmunds. Por lo tanto, Dave Edmunds no debe eliminarse; de lo contrario, faltará la información requerida para *I hear you knocking*. Sin embargo, Listado le permite elegir un registro diferente en lugar de Dave Edmunds.



Figura 2

En el formulario hay un filtro incorporado cuya activación puede indicarle qué categorías no son necesarias en la tabla *Media* (Ver Figura 3). En el caso que se acaba de describir, casi todos los autores de ejemplo están en uso. Solo el registro de Erich Kästner se puede eliminar sin consecuencias para ningún otro registro en los medios.



Figura 3

El filtro está codificado en este caso. Se encuentra en las propiedades del formulario (Ver Figura 4). Dicho filtro se activa automáticamente cuando se inicia el formulario. Se puede apagar y encender. Si se elimina, se puede acceder nuevamente mediante una recarga completa del formulario.

Esto implica más que solo actualizar los datos; todo el documento del formulario debe cerrarse y luego volverse a abrir.

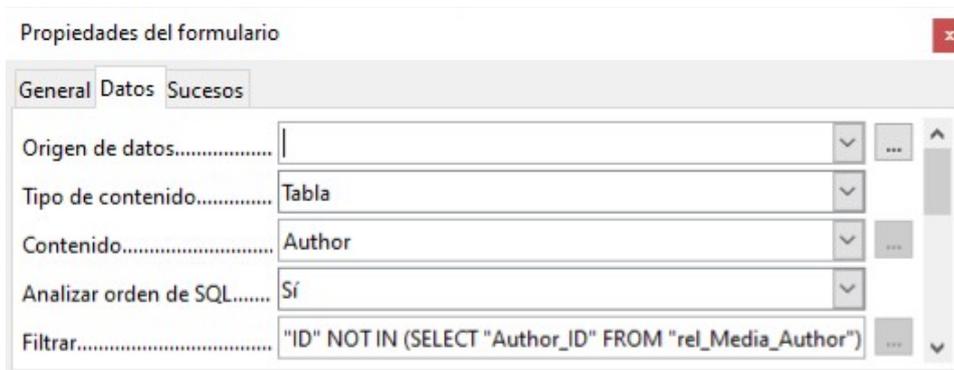


Figura 4

## Consultas para encontrar entradas huérfanas

El filtro anterior es parte de una consulta que se puede usar para buscar entradas huérfanas.

```
SELECT "Surname", "Firstname" FROM "Author" WHERE "ID" NOT IN (SELECT "Author_ID"
FROM "rel_Media_Author")
```

Si una tabla contiene claves externas de otras tablas, la consulta debe ampliarse en consecuencia. Esto afecta, por ejemplo, a la tabla *Town*, que tiene claves foráneas tanto en la tabla *Media* como en la tabla *Postcode*. Por lo tanto, los registros de la tabla *Town* que se van a eliminar no deben referenciarse en ninguna de estas tablas. Esto está determinado por la siguiente consulta:

```
SELECT "Town" FROM "Town" WHERE "ID" NOT IN (SELECT "Town_ID" FROM "Media") AND
"ID" NOT IN (SELECT "Town_ID" FROM "Postcode")
```

Las entradas huérfanas se pueden eliminar seleccionando todas las entradas que pasan el filtro establecido y utilizando la opción *Eliminar* en el menú contextual del puntero de registro, que se activa haciendo clic con el botón derecho.

## Velocidad de búsqueda en base de datos

---

### Efecto de consultas

Son solo estas consultas, utilizadas en la sección anterior para filtrar datos, las que resultan insatisfactorias con respecto a la velocidad máxima de búsqueda en una base de datos. El problema es que en bases de datos grandes, la subconsulta recupera una cantidad de datos grande con la que se debe comparar cada registro visualizable. Solo las comparaciones con la relación IN permiten comparar un solo valor con un conjunto de valores. La consulta:

```
... WHERE "ID" NOT IN (SELECT "Author_ID" FROM "rel_Media_Author")
```

puede contener una gran cantidad de claves foráneas posibles de la tabla *rel\_Media\_Author*, que primero debe compararse con las claves primarias de la tabla *Authors* para cada registro de esa tabla. Por lo tanto, dicha consulta no es adecuada para el uso cotidiano, pero puede ser necesaria para el mantenimiento de la base de datos. Para el uso diario, las funciones de búsqueda deben construirse de manera diferente para que la búsqueda de datos no sea demasiado larga y no dañe el trabajo con la base de datos.

### Efecto de Listados y Cuadros combinados.

Cuanto más controles de formulario de listados estén integrados en un formulario, y cuanto más datos contengan, más tardará en cargarse el formulario, ya que estos listados deben crearse.

Cuanto mejor configure la interfaz gráfica para que inicialmente lea el contenido del listado solo parcialmente, menos demora habrá.

Los listados se crean mediante consultas, y estas consultas deben ejecutarse cuando se carga el formulario para cada listado.

La misma estructura de consulta para más listados se realiza mejor usando una vista común, en lugar de crear campos con la misma sintaxis usando repetidamente los órdenes SQL almacenados en los listados. Las vistas son, sobre todo, preferibles para sistemas de bases de datos externas, ya que aquí el servidor se ejecuta significativamente más rápido que una consulta que debe ser reunida por la GUI y recién puesta en el servidor. El servidor trata las vistas como consultas locales completas.

### Influencia del sistema de base de datos utilizado

La base de datos interna HSQLDB está configurada para garantizar que Base y Java funcionen bien juntos.

Desafortunadamente, a partir de la versión 3.5 de LibreOffice, Base tuvo problemas con la velocidad de procesamiento precisamente en esta área. Esto se volvió particularmente notable cuando se trataba de tablas grandes con varios miles de registros. Estos problemas tuvieron varias causas y solo desaparecieron con la versión 4.1.

Las bases de datos externas se ejecutan significativamente más rápido. Las conexiones directas a MySQL o PostgreSQL y las conexiones que utilizan ODBC, se ejecutan prácticamente a la misma velocidad. JDBC también depende de la cooperación con Java, pero aún funciona más rápido que una conexión interna usando HSQLDB.