



Guía de Calc

Capítulo 13 *Calc como base de datos* *simple*

Una guía para usuarios y programadores de macros

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

Zachary Parliman
Carlos Díez Izarra
Juan C. Sanz

Jean Hollis Weber
Milton Tirado Garcia

Kees Kriek
Jose Maria Lopez Saez

De ediciones previas

Andrew Pitonyak
Simon Brydon

Barbara Duprey

Jean Hollis Weber

Comentarios y sugerencias

Puede dirigir cualquier clase de comentario o sugerencia acerca de este documento a: 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 borrada

Fecha de publicación y versión del programa

Versión en español publicada el 23 de abril 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.

<i>Windows o Linux</i>	<i>Equivalente en Mac</i>	<i>Efecto</i>
Herramientas > Opciones opción de menú	LibreOffice > Preferencias	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

Contents

Derechos de autor	2
Colaboradores.....	2
De esta edición.....	2
De ediciones previas.....	2
Comentarios y sugerencias.....	2
Fecha de publicación y versión del programa.....	2
Uso de LibreOffice en macOS	2
Introducción	5
Una base de datos.....	5
Calc como un programa similar a una base de datos.....	6
Asociar un intervalo con un nombre	6
Intervalos con Nombre.....	6
Crear intervalos con nombre con macros.....	9
Usar referencias relativas con expresiones nombradas.....	10
Crear intervalos con nombres usando encabezados de fila o columna.....	11
Creación de intervalos con nombres desde etiquetas usando macros.....	12
Intervalos de Base de Datos.....	14
Creación de intervalos de base de datos con macros.....	16
Ordenar	17
Ordenar una tabla usando una columna con una macro.....	18
Ordenar una tabla utilizando múltiples columnas.....	19
Recuperar información ordenada de un intervalo.....	20
Filtrado	21
Filtro automático.....	21
Conmutar Filtros Automáticos con una macro.....	23
Crear filtros estándar con macros.....	23
Limpiar todos los filtros de una hoja de cálculo.....	25
Filtrar múltiples columnas y filtrar con expresiones regulares.....	26
Filtros avanzados.....	27
Ejemplo de filtro avanzado.....	28
Usar un filtro avanzado con macros.....	28
Copiar los resultados de filtros avanzados a ubicaciones diferentes.....	30
Funciones útiles similares a bases de datos	31
Funciones específicas de base de datos.....	33

Introducción

Aunque este es un programa de hoja de cálculo, Calc tiene suficiente funcionalidad para actuar como una plataforma sencilla, pero capaz, similar a una base de datos. Este capítulo presenta una perspectiva general de estas capacidades, y las explica mediante la utilización de ejemplos tanto de macros, como de GUI (interfaz gráfica de usuario).



Nota

Aunque fue creada para programadores de macros, esta guía está pensada para ser accesible a todos los usuarios. Si no desea utilizar macros, sencillamente omita las secciones que se ocupan de ellas. Sin embargo, si está interesado en aprender más sobre ellas, consulte el *Capítulo 12, Macros de Calc*, en este libro; así como el libro de Andrew Pitonyak: *OpenOffice.org Macros Explicadas (OOME)*.

Toda la información sobre macros en este capítulo ha sido extraída, o adaptada de la referencia de OOME, y de la API de LibreOffice en <https://api.libreoffice.org/docs/idl/ref/index.html>.

Una base de datos

En una base de datos típica, los datos relacionados se organizan en tablas, que se ordenan en una serie de filas y columnas de tipo matricial, similares a una hoja de cálculo. Cada fila de una tabla representa un registro de datos, mientras que cada columna representa a un campo dentro de cada registro. Cada celda en un campo contiene un dato individual o atributo, como un nombre, mientras que cada registro consta de atributos relacionados, que se corresponden con una entidad única, como una persona. Una tabla de una base de datos suele tener un número determinado de campos, pero puede tener un número de registros indefinido.

Si bien una tabla puede tener cientos o miles de filas, los registros individuales se pueden encontrar, recuperar y actualizar fácilmente mediante la utilización de peticiones de información, denominadas consultas, que buscan los registros que satisfacen un conjunto de criterios específicos. Es esta facilidad de acceso la que hace a una tabla de base de datos más útil que simplemente archivar información en una hoja de cálculo desordenada.

Para ilustrar este concepto de tabla de base de datos, considere el ejemplo de una hoja de calificación de una clase (*Figura 1*). En esta hoja cada fila representa a cada uno de los estudiantes que asisten a la clase, mientras que cada columna contiene sus nombres y calificaciones. Con esta tabla, puedes buscar rápidamente calificaciones de cada estudiante, buscando simplemente por sus nombres, y puedes determinar qué estudiantes aprueban la asignatura, filtrando registros con puntuaciones medias de suspensos.

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
2	Andrew	90	100	82	90	88	95	90.33
3	Bethany	95	100	82	80	88	93	89.67
4	Charles	80	93	73	80	75	84	80.83
5	David	75	86	91	40	88	79	76.5
6	Emily	100	100	81	100	75	94	91.67
7	Ferdinand	85	93	73	60	50	72	72.17
8	Geogia	70	80	55	39	75	67	64.33
9	Harley	85	93	82	70	75	76	80.17
10	Ian	100	100	91	90	100	96	96.17
11	Jennifer	85	93	73	80	100	90	86.83

Figura 1: Ejemplo de hoja de calificaciones



Nota

Este diseño tabular simple se basa en el modelo de base de datos relacional, que es uno de los modelos de diseño más comunes y conocidos utilizados en las bases de datos modernas.

Calc como un programa similar a una base de datos

Como se ha mencionado, una tabla de base de datos es similar a una hoja de cálculo, e incluso puede estar contenida dentro de una. Además, como programa de hoja de cálculo, Calc ofrece varias características, en particular la clasificación y el filtrado, que permiten a los usuarios buscar en tablas, de forma similar a como se haría en un programa de base de datos como LibreOffice Base, o Microsoft Access. Si bien esto no convierte a Calc en un sustituto de ninguno de esos programas, sigue siendo útil para administrar datos en un contexto personal o profesional a pequeña escala, sin tener que aprender a usar un sistema de base de datos dedicado.

Asociar un intervalo con un nombre

Para configurar una tabla de base de datos en una hoja de cálculo de Calc, primero debe configurar un área para que la ocupe. Esto es necesario, ya que algunas de las características de base de datos de Calc dependen del acceso o modificación de la ubicación de una tabla. Esta área está representada por un intervalo, que es un grupo contiguo de una o más celdas. Para facilitar el acceso al intervalo de una tabla, puede asignarle un nombre significativo. Hacer esto tiene cuatro beneficios particulares:

- **Darle un nombre a un intervalo hace que sea más fácil de identificar**, especialmente si está trabajando con múltiples intervalos en un documento.
- **Un intervalo con nombre puede ser referenciado por su nombre, y no solo por su dirección**. Por ejemplo, si tiene un intervalo llamado *Puntuaciones*, simplemente puede hacer referencia a él en una celda con una ecuación como =SUMA (Puntuaciones).
- **Las referencias por nombre a un intervalo con nombre, se actualizan automáticamente cada vez que se cambia la dirección del intervalo**. Esto evita la necesidad de cambiar referencias individuales cada vez que se modifica la ubicación de un intervalo.
- **Todos los intervalos con nombre se pueden ver y acceder rápidamente a través del navegador**, que se abre presionando la tecla *F5*, o haciendo clic en el icono  en el panel de la *Barra lateral*.

Existen dos tipos de intervalo con nombre en Calc: intervalos de bases de datos, que almacenan configuraciones para operaciones similares a las bases de datos, e intervalos con nombre estándar, que no.

Intervalos con Nombre

Los intervalos con nombre estándar se crean utilizando el cuadro de diálogo *Definir nombre* (Figura 2), que se abre seleccionando **Hoja > Intervalos y expresiones con nombre > Definir** en la barra de *Menú*.

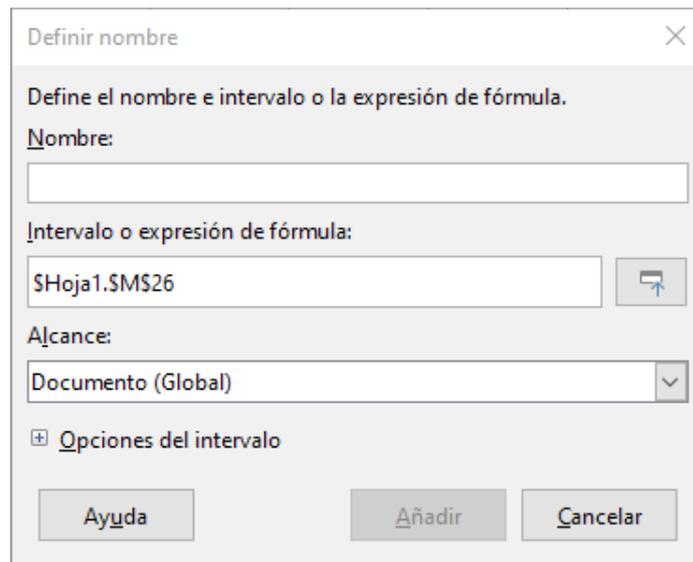


Figura 2: Cuadro de diálogo Definir nombre

Para crear un intervalo con nombre, seleccione un intervalo de celdas de una hoja de cálculo, y luego abra el cuadro de diálogo. A continuación, asigne un nombre significativo al intervalo, y haga clic en **Añadir** para agregarlo a la lista de intervalos con nombre del documento actual. Luego, puede acceder y modificar estos intervalos utilizando el cuadro de diálogo *Gestionar nombres*, que se abre seleccionando **Hoja > Intervalos y expresiones con nombre > Gestionar** en la barra de menú (Figura 3), o pulsando *Ctrl+F3*. Para obtener más detalles sobre cómo crear y administrar intervalos, consulte el *Capítulo 6, Impresión, exportación, correo electrónico y firma* y el *Capítulo 7, Fórmulas y funciones*.

Gestionar nombres ✕

Nombre	Intervalo o expresión de fórmula	Alcance

Nombre:

Intervalo o expresión de fórmula: ➔

Alcance: Documento (Global) ▾

⊕ Opciones del intervalo

Añadir Eliminar

Ayuda
Aceptar
Cancelar

Gestionar nombres ✕

Nombre	Intervalo o expresión de fórmula	Alcance

Nombre:

Intervalo o expresión de fórmula: ➔

Alcance: Documento (Global) ▾

⊕ Opciones del intervalo

Añadir Eliminar

Ayuda
Aceptar
Cancelar

Figura 3: Cuadro de diálogo Gestionar nombres

Crear intervalos con nombre con macros

En una macro, se accede a un intervalo con nombre, que se crea y se elimina mediante el servicio **NamedRanges** de un documento Calc. Este servicio tiene varios métodos asociados, los siguientes son particularmente útiles para crear rangos con nombre:

getByName(Nombre)

Devuelve el rango o expresión con el nombre especificado.

getElementNames()

Devuelve una matriz de todos los intervalos con nombre en el documento actual.

hasByName(Name)

Devuelve un booleano: VERDADERO si un intervalo con este nombre está en el documento actual, y FALSO en caso contrario.

addNewByName(Name, Content, Position, Type)

Agrega un nuevo rango con nombre al documento actual. Este método tiene cuatro argumentos:

- *Nombre (Name)* – Una cadena que contiene el nombre del nuevo intervalo
- *Contenido (Content)* – Una cadena que contiene la dirección de rango o la expresión de fórmula que se nombra
- *Posición (Position)* – La dirección base para referencias de celda relativas
- *Tipo (Type)* – Una combinación de indicadores que especifican el tipo de rango con nombre que se está definiendo. Estos indicadores se enumeran en la *Tabla 1*. Este parámetro tiene el valor predeterminado cero para cualquier rango con nombre común.

Tabla 1: Referencia de grupo constante com.sun.star.sheet.NamedRangeFlag

Valor	Nombre	Descripción
1	FILTER_CRITERIA	El intervalo contiene criterios de filtro.
2	PRINT_AREA	El intervalo se puede utilizar como rango de impresión.
4	COLUMN_HEADER	El intervalo se puede utilizar como encabezados de columna para imprimir.
8	ROW_HEADER	El intervalo se puede utilizar como encabezados de fila para imprimir.

Como ejemplo, la macro en el *Listado 1* usa los métodos anteriores para verificar si existe un intervalo con un nombre dado. Si no existe, la macro crea un intervalo con el nombre, y lo configura para acceder al intervalo de celdas B3: D6.

Listado 1: AddNamedRange crea un nuevo intervalo con nombre que hace referencia a \$Sheet1.\$B\$3:\$D\$6

```
Sub AddNamedRange()  
    Dim oRange      ' El intervalo creado.  
    Dim oRanges     ' Todos los intervalos con nombre.  
    Dim sName$     ' Nombre del intervalo con nombre que se creará.  
    Dim oCell      ' Objeto de celda.  
    Dim s$  
  
    sName$ = "MyNRange"  
    oRanges = ThisComponent.NamedRanges
```

```

If NOT oRanges.hasByName(sName$) Then
    REM Configuración de la dirección base para referencias de celda relativas
    Dim oCellAddress As new com.sun.star.table.CellAddress
    oCellAddress.Sheet = 0      'La primera hoja.
    oCellAddress.Column = 1    'Columna B.
    oCellAddress.Row = 2      'Fila 3.

    REM El primer argumento es el nombre del intervalo.
    REM El segundo argumento es una cadena que define la fórmula
    REM o expresión que se utilizará.
    REM El tercer argumento especifica la dirección base para
    REM las referencias de celda relativas.
    REM El cuarto argumento es un conjunto de indicadores que definen
    REM cómo se usa el intervalo, pero la mayoría de los intervalos usan 0
    REM El cuarto argumento usa valores de las constantes NamedRangeFlag.
    s$ = "$Sheet1.$B$3:$D$6"
    oRanges.addNewByName(sName$, s$, oCellAddress, 0)
End If
REM Obtener el intervalo con nombre creado.
oRange = ThisComponent.NamedRanges.getByNamedRange(sName$)

REM Imprime la cadena contenida en la celda $Sheet1.$B$3
oCell = oRange.getReferredCells().getCellByPosition(0,0)
Print oCell.getString()
End Sub

```

Usar referencias relativas con expresiones nombradas

Si un intervalo con nombre utiliza cualquier dirección de celda que no sea absoluta, entonces se hará referencia a estas direcciones en relación con la dirección base del intervalo, que se define mediante el tercer argumento del método **addNewByName**, *Posición*. Este comportamiento se ilustra en el *Listado 2*, donde la macro `AddNamedFunction` crea la expresión nombrada `AddLeft`. Esta expresión hace referencia a la ecuación `A3 + B3`, con `C3` como su dirección base. Debido a que se están utilizando referencias relativas, `AddLeft` suma los valores de las dos celdas directamente a la izquierda de cualquier celda que contenga la ecuación `=AddLeft`. Por ejemplo, si se hace referencia a `AddLeft` en la celda `E5`, entonces sumará los valores en `C5` y `D5` (*Figura 4*).



Nota

Para obtener más información sobre referencias absolutas y relativas, consulte el [Capítulo 7, Fórmulas y funciones](#).

Listado 2: AddNamedFunction crea la fórmula llamada AddLeft

```

Sub AddNamedFunction()
    Dim oSheet          'Hoja que contiene el intervalo.

```

```

Dim oCellAddress 'Dirección para referencias relativas.
Dim oRanges      'La propiedad NamedRanges.
Dim oRange       'Intervalo de celda individual.
Dim sName As String 'Nombre de la ecuación a crear.

```

```

sName = "AddLeft"
oRanges = ThisComponent.NamedRanges
If NOT oRanges.hasByName(sName) Then
    oSheet = ThisComponent.getSheets().getByIndex(0)
    oRange = oSheet.getCellRangeByName("C3")
    oCellAddress = oRange.getCellAddress()
    oRanges.addNewByName(sName, "A3+B3", oCellAddress, 0)
End If
End Sub

```

	A	B	C	D	E
5			12	30	42
6					

Figura 4: AddLeft suma los valores desde C5 a D5, en E5



Consejo

El listado 2 muestra otro atributo poco conocido de Calc: los intervalos con nombres son un subconjunto de las expresiones con nombre, las cuales también pueden contener fórmulas.

Crear intervalos con nombres usando encabezados de fila o columna

Con la herramienta *Crear Nombres*, a la cual se accede seleccionando **Hoja > Intervalos y expresiones con nombre > Crear** desde la barra de menú (Figura 5), puedes crear varios intervalos con nombres de manera simultánea, desde los encabezados de una tabla. Estos encabezados pueden ser extraídos de los límites de la tabla – filas superior e inferior, y columnas izquierda y derecha – y cada fila o columna que se corresponde con cada encabezado, se utiliza por sí mismo para crear los intervalos con nombres. Por ejemplo, si elige crear intervalos desde los encabezados contenidos en la fila superior de una tabla, cada rango será generado desde las columnas individuales que se corresponden con cada etiqueta de encabezado.



Nota

Las celdas de encabezado no son incluidas en los intervalos con nombres generados utilizando la herramienta *Crear nombres*. Esto se debe a que las etiquetas en cada una de estas celdas son empleadas para dar nombre a los rangos.

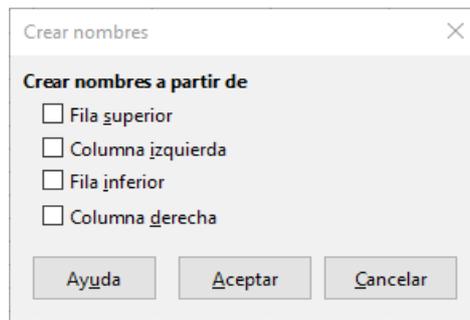


Figura 5: Cuadro de diálogo Crear nombres

Para utilizar la herramienta *Crear nombres*:

- 1) En una hoja de cálculo, seleccione la tabla desde la cual generar el intervalo con nombres. Asegúrese de incluir las filas o columnas de encabezados como parte de su selección.
- 2) Abra el cuadro de diálogo *Crear nombres* seleccionando **Hoja > Intervalos y expresiones con nombre > Crear** desde la barra de menú.
- 3) Calc automáticamente identifica qué filas o columnas contienen encabezados, y marcará las casillas de verificación – Fila superior, Columna izquierda, Fila inferior, Columna derecha – que proceda. No obstante, si desea cambiar esta selección, puede seleccionar manualmente cualquiera de las casillas en este momento.
- 4) Haga clic en **Aceptar** para cerrar el cuadro de diálogo y crear un nuevo intervalo de nombres.

Creación de intervalos con nombres desde etiquetas usando macros.

Además del cuadro de diálogo *Crear nombres*, los intervalos con nombres se pueden generar desde etiquetas, utilizando macros, mediante el método **addNewFromTitles**.

addNewFromTitles(Origen, Límite)

Crea intervalos de nombres desde los encabezados de columna o fila. Este método tiene dos argumentos:

- *Origen* – La dirección de la celda de intervalo, del intervalo de nombres a crear.
- *Límite* – Es un valor que especifica la localización de las etiquetas de los encabezados. Este valor es uno entre cuatro posibles valores de una lista (*Tabla 2*):

Tabla 2: lista de valores com.sun.star.sheet.Border

Lista	Descripción
TOP	Selecciona la fila límite superior.
BOTTOM	Selecciona la fila límite inferior.
RIGHT	Selecciona la columna límite derecha.
LEFT	Selecciona la columna límite izquierda.



Consejo

Para generar nombres desde varios límites, debe usar **addNewFromTitles** para cada fila o columna encabezado que desea utilizar.

La macro en el *Listado 3* crea tres intervalos de nombres, utilizando los encabezados de la fila superior A1:C20 (*Figura 6*). La *Figura 7* muestra los intervalos resultantes, listados en el cuadro de

diálogo *Gestionar nombres*, al cual se accede seleccionando **Hoja > Intervalos y expresiones con nombres > Gestionar**.

	A	B	C
1	X	X^2	X^3
2	1	1	1
3	2	4	8
4	3	9	27
5	4	16	64
6	5	25	125
7	6	36	216
8	7	49	343
9	8	64	512
10	9	81	729
11	10	100	1000
12	11	121	1331
13	12	144	1728
14	13	169	2197
15	14	196	2744
16	15	225	3375
17	16	256	4096
18	17	289	4913
19	18	324	5832
20	19	361	6859

Figura 6: Intervalo ejemplo A1:C20

Listado 3: AddManyNamedRanges crea intervalos de nombres utilizando columnas etiquetadas

```
Sub AddManyNamedRanges()  
    Dim oSheet      'Hoja que contiene el intervalo de nombres.  
    Dim oAddress    'Intervalo de direcciones.  
    Dim oRanges     'La propiedad NamedRanges.  
    Dim oRange      'Intervalo de celda individual.  
  
    oRanges = ThisComponent.NamedRanges  
    oSheet = ThisComponent.getSheets().getByIndex(0)  
  
    oRange = oSheet.getCellRangeByName("A1:C20")  
    oAddress = oRange.getRangeAddress()  
    oRanges.addNewFromTitles(oAddress, com.sun.star.sheet.Border.TOP)  
End Sub
```

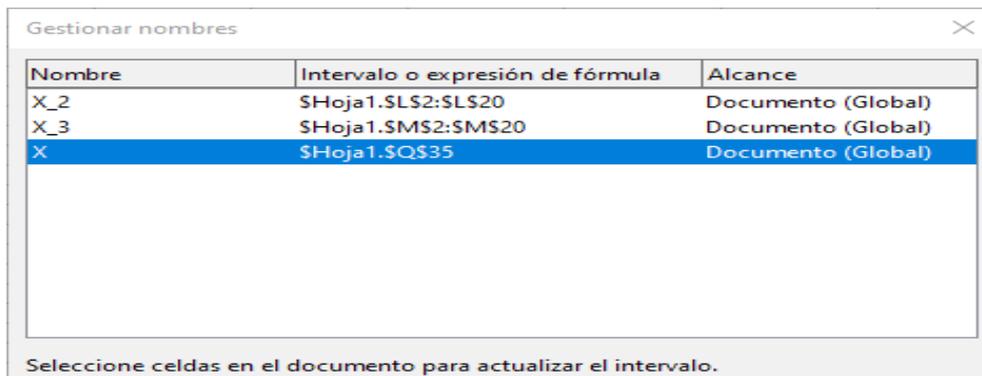


Figura 7: Cuadro de diálogo Gestionar Nombres, con intervalos de nombres ya generados.



Atención

Evite asignar la misma etiqueta a varias filas o columnas, ya que del mismo modo, los intervalos que se generen a partir de ellos, compartirán el mismo nombre y pueden terminar siendo sobrescritos por Calc.

Intervalos de Base de Datos

Aunque puede utilizarse como un intervalo con nombre normal, un intervalo de base de datos está pensado para usarse, como es lógico, como una tabla de base de datos, en la que cada fila representa un registro, y cada celda un campo dentro de cada registro. Concretamente, un intervalo de base de datos se diferencia de un intervalo con nombre en lo siguiente:

- Un intervalo de base de datos no puede ser una expresión de tipo fórmula, solo un intervalo de celdas. Este intervalo puede ser formateado como una tabla, con la primera fila reservada para encabezados y la última fila para subtotales. El formato de las celdas también puede preservarse para cada campo en la tabla.
- Los intervalos de base de datos no se pueden referenciar de manera relativa a una dirección base dentro de una hoja de cálculo.
- Los intervalos de base de datos almacenan los ajustes de importación de datos, de ordenación, de filtrado y subtotales, en estructuras de datos llamadas descriptores, los cuales se pueden recuperar, y a los que se puede acceder utilizando macros.
- Un intervalo de base de datos se puede vincular a una fuente externa de base de datos y se puede refrescar mediante la selección **Datos > Actualizar intervalo** desde la barra de menú. Más adelante, en el *Capítulo 10, Vinculando Datos en Calc*, se explica con más detalle cómo registrar y vincular a fuentes de bases de datos externas.
- Los intervalos de bases de datos se pueden crear, modificar y borrar, mediante el uso del cuadro de diálogo *Definir Intervalo de base de datos*, el cual se puede abrir seleccionando **Datos > Definir intervalo** desde la barra de menú (*Figura 8*).

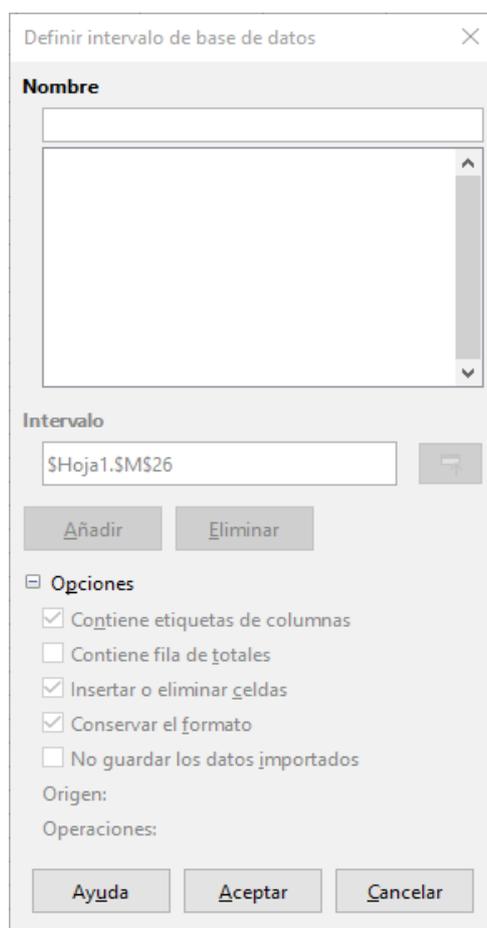


Figura 8: Cuadro de dialogo Definir Intervalo de base de datos

Para crear un intervalo de base de datos:

- 1) Seleccione un intervalo de celdas de una hoja de cálculo.
- 2) Abra el cuadro de diálogo *Definir intervalo de base de datos* utilizando **Datos > Definir intervalo**.
- 3) Escriba un nombre para el intervalo en el campo *Nombre*. Utilice únicamente letras, números y barras bajas; no están permitidos los espacios, guiones y otros caracteres.
- 4) Pulse sobre el signo más (+) que se encuentra junto a *Opciones*, para expandir esta sección, y así poder ver y seleccionar las opciones siguientes:
 - *Contiene etiquetas de columnas* – Indica si la fila superior se reserva para campos de encabezados.
 - *Insertar o eliminar celdas* – Si está activa, esta opción insertará nuevas filas y columnas en el intervalo de base de datos cuando se añadan nuevos registros a su origen. Solo funciona si al intervalo se vincula una base de datos externa. Para actualizar de manera manual el intervalo de base de datos, utilice **Datos > Actualizar intervalo**.
 - *Conservar el formato* – Aplica los formatos de celda existentes en la primera fila de datos, a todo el intervalo de base de datos.
 - *No guardar los datos importados* – Si se selecciona, esta opción únicamente graba una referencia a la base de datos origen; los contenidos de las celdas de intervalo no se preservan.

- *Origen* – Muestra información sobre el origen de la base de datos actual, si es que existe.
 - *Operaciones* – Indica qué operaciones, tales como ordenación o filtrado, han sido aplicadas al intervalo de base de datos.
- 5) Haga clic en **Añadir** para incorporar un intervalo a la lista de intervalos de base de datos mostrada bajo el campo *Nombre*.

Para modificar un intervalo de base de datos existente:

- 1) Seleccione un intervalo de la lista de rangos mostrada bajo el campo *Nombre*, o escriba su nombre en el campo *Nombre*. El botón *Añadir* cambiará a *Modificar* en este punto.
- 2) Realice cualquier modificación en el campo *Intervalo* y en la sección de *Opciones*.
- 3) Haga clic en **Modificar** para actualizar el intervalo.

Para borrar un intervalo de base de datos existente, seleccione dicho rango de la lista de rangos mostrada, y pulse **Eliminar**.

Para seleccionar un intervalo de base de datos existente en el documento actual, abra el cuadro de diálogo *Seleccionar intervalo de base de datos* mediante **Datos > Seleccionar intervalo** desde la barra de menú (Figura 9). A continuación, seleccione un intervalo de la lista de *Intervalos*, y pulse **Aceptar**. Calc resaltará automáticamente la posición del intervalo en la hoja de cálculo en la que se encuentra ubicado.

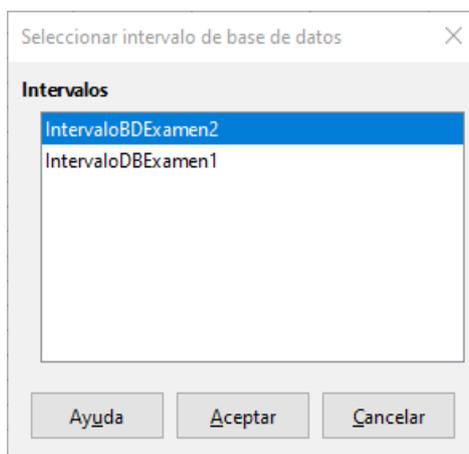


Figura 9: Cuadro de dialogo
Seleccionar intervalo de base de datos

Creación de intervalos de base de datos con macros

En una macro, mediante la utilización del servicio **DatabaseRanges**, se puede crear, acceder y borrar un intervalo de base de datos. Este servicio presenta muchos de los mismos métodos contenidos en el servicio **NamedRanges**, pero carece del método **addNewFromTitles**. **DatabaseRanges** también utiliza una versión reducida del método **addNewByName**, que carece de los argumentos para una dirección base relativa, y tipo de intervalo:

addNewByName(Name, Range)

Añade un intervalo de base de datos nuevo, al documento actual.

Como ejemplo de cómo crear un intervalo mediante la utilización de este método, la macro en el *Listado 4* crea un intervalo de base de datos denominada *MyName*, y aplica filtros automáticos a cada una de las columnas del intervalo:

Listado 4: AddNewDatabaseRange crea un intervalo de base de datos y aplica un filtro automático.

```

Sub AddNewDatabaseRange()
    Dim oRange 'objeto DatabaseRange.
    Dim oAddr 'Intervalo de dirección de celda para intervalo de base de datos.
    Dim oSheet 'Primera hoja, la cual contiene el intervalo.
    Dim oDoc 'Referencia ThisComponent con un nombre más corto.

    oDoc = ThisComponent
    If NOT oDoc.DatabaseRanges.hasByName("MyName") Then
        oSheet = ThisComponent.getSheets().getByIndex(0)
        oRange = oSheet.getCellRangeByName("A1:F10")
        oAddr = oRange.getRangeAddress()
        oDoc.DatabaseRanges.addNewByName("MyName", oAddr)
    End If
    oRange = oDoc.DatabaseRanges.getByName("MyName")
    oRange.AutoFilter = True
End Sub

```

Ordenar

Ordenar consiste en el proceso de reorganizar los datos en un intervalo de una hoja de cálculo, de acuerdo con un criterio de orden especificado. En Calc, el proceso de ordenar se realiza normalmente utilizando el cuadro de diálogo *Ordenación*, al cual se accede mediante la selección **Datos > Ordenar** desde la barra de menú. El cómo usar este cuadro de diálogo y sus opciones, se describe con detalle en el *Capítulo 2, Introducir, Editar y Formatear Datos*. Aquí, presentamos cómo ordenar datos en el contexto de las macros.

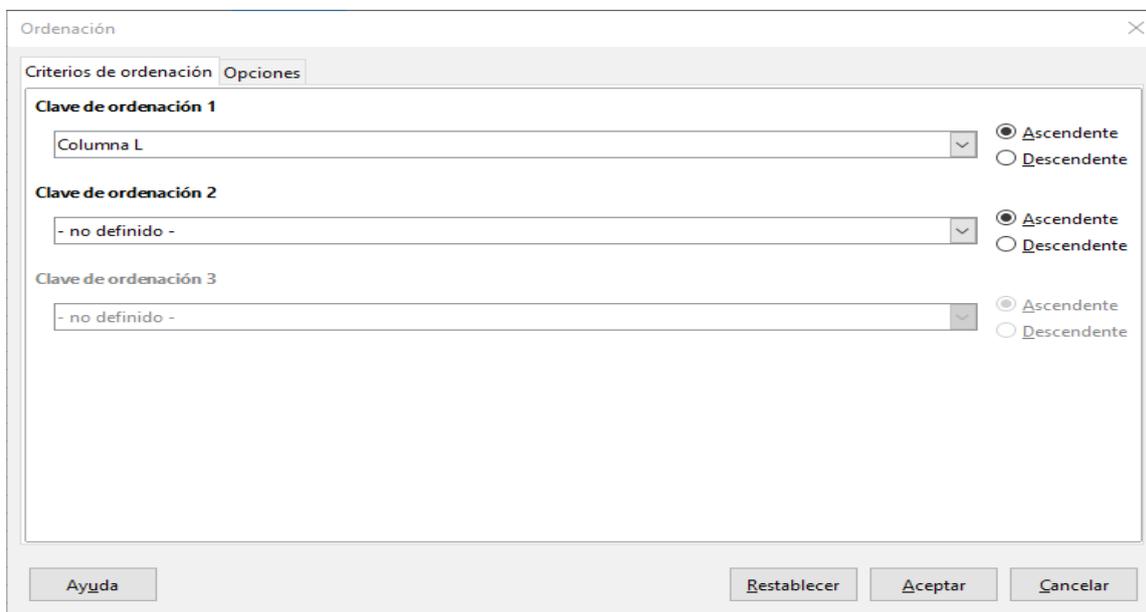


Figura 10: Cuadro de dialogo Ordenación

Ordenar una tabla usando una columna con una macro

En una macro de Calc, los datos contenidos en un intervalo, se ordenan llamando al método **sort()** sobre el objeto intervalo. Cuando una operación ordenar se ejecuta sobre un intervalo, un conjunto de propiedades conocidas como descriptor de ordenación, se transmite a la rutina de ordenar. Las propiedades del descriptor contienen los campos a ordenar, que son las estructuras que informan a Calc cómo ordenar, de acuerdo con los datos contenidos dentro de una de sus filas o columnas.

Como ejemplo, la macro en el *Listado 5* ordena las calificaciones de la *Figura 1* de acuerdo con la calificación media, en orden descendente. Los resultados se muestran en la *Figura 11*.

Listado 5: SortAverageGrade ordena el intervalo de datos (A1:H11) de la hoja de calificaciones utilizando una única columna

```
Sub SortAverageGrade
  Dim oSheet
  Dim oRange
  Dim oSortFields(0) as new com.sun.star.util.SortField
  Dim oSortDesc(0) as new com.sun.star.beans.PropertyValue

  oSheet = ThisComponent.Sheets(0)
  REM Establece el intervalo a ordenar
  oRange = oSheet.getCellRangeByName("A1:H11")

  REM Ordena por el campo de calificación media en el intervalo,
  REM en orden descendente.
  oSortFields(0).Field = 7
  oSortFields(0).SortAscending = FALSE

  REM Establece los campos de ordenación a usar
  oSortDesc(0).Name = "SortFields"
  oSortDesc(0).Value = oSortFields()

  REM ¡Ahora ordena el intervalo!
  oRange.Sort(oSortDesc())
End Sub
```

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
2	Ian	100	100	91	90	100	96	96.17
3	Emily	100	100	81	100	75	94	91.67
4	Andrew	90	100	82	90	88	95	90.33
5	Bethany	95	100	82	80	88	93	89.67
6	Jennifer	85	93	73	80	100	90	86.83
7	Charles	80	93	73	80	75	84	80.83
8	Harley	85	93	82	70	75	76	80.17
9	David	75	86	91	40	88	79	76.5
10	Ferdinand	85	93	73	60	50	72	72.17
11	Geogia	70	80	55	39	75	67	64.33

Figura 11: Hoja de calificaciones después de ordenar por la calificación media en orden descendente.

Ordenar una tabla utilizando múltiples columnas

Como con el cuadro de diálogo *Ordenar*, un intervalo se puede ordenar con una macro, utilizando hasta tres columnas o filas. Ordenar con columnas o filas adicionales, es tan fácil como añadir más campos de ordenación al descriptor de ordenación. La macro en el *Listado 6* utiliza nuevamente la hoja de clasificaciones ejemplo de la *Figura 1*, para mostrar cómo ordenar por dos columnas. La *Figura 12* muestra los resultados de esta operación – tenga en cuenta que los registros son ordenados en primer lugar por las puntuaciones del *Examen 1*, y después, por las puntuaciones del *Examen 2*.

Listado 6: `SortByQuizScores` ordena el intervalo de datos (A1:H11) de la hoja de puntuaciones utilizando dos columnas

Sub `SortByQuizScores`

```
Dim oSheet
```

```
Dim oRange
```

```
Dim oSortFields(1) as new com.sun.star.util.SortField
```

```
Dim oSortDesc(0) as new com.sun.star.beans.PropertyValue
```

```
oSheet = ThisComponent.Sheets(0)
```

```
REM Establece el intervalo sobre el cual ordenar
```

```
oRange = oSheet.getCellRangeByName("A1:H11")
```

```
REM Ordena por el campo Examen 1 en el intervalo
```

```
oSortFields(0).Field = 4
```

```
oSortFields(0).SortAscending = True
```

```
oSortFields(0).FieldType = com.sun.star.util.SortFieldType.NUMERIC
```

```
REM Ordena por el campo Examen 2 en el intervalo
```

```
oSortFields(1).Field = 5
```

```
oSortFields(1).SortAscending = True
```

```

oSortFields(1).FieldType = com.sun.star.util.SortFieldType.ALPHANUMERIC

REM Establece los campos de ordenación a usar
oSortDesc(0).Name = "SortFields"
oSortDesc(0).Value = oSortFields()

REM ¡Ahora ordena el intervalo!
oRange.Sort(oSortDesc())
End Sub

```

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
2	Georgia	70	80	55	39	75	67	64.33
3	David	75	86	91	40	88	79	76.5
4	Ferdinand	85	93	73	60	50	72	72.17
5	Harley	85	93	82	70	75	76	80.17
6	Charles	80	93	73	80	75	84	80.83
7	Jennifer	85	93	73	80	100	90	86.83
8	Bethany	95	100	82	80	88	93	89.67
9	Andrew	90	100	82	90	88	95	90.33
10	Ian	100	100	91	90	100	96	96.17
11	Emily	100	100	81	100	75	94	91.67

Figura 12: Hoja de calificaciones ordenada por puntuaciones de exámenes en orden ascendente

Recuperar información ordenada de un intervalo

Puede utilizar el método **createSortDescriptor()** para recuperar la información ordenada, para un determinado intervalo de celdas. Si este método es llamado sobre un intervalo de base de datos, creará un descriptor de ordenación, utilizando la información ordenada almacenada con ese intervalo. Por otra parte, si se llama al método **createSortDescriptor()** sobre un intervalo con nombres estándar o sobre un intervalo sin nombres, generará un descriptor de ordenación con las propiedades por defecto. En cualquier caso, el descriptor de ordenación nuevo que se genera se puede modificar, y pasar como un argumento a una rutina de ordenación llamada en un intervalo determinado.

La macro en el *Listado 7* demuestra cómo generar y mostrar la información ordenada asociada con un intervalo. La salida que se genera por esta macro, se muestra en la *Figura 13*.

Listado 7: DisplaySortDescriptor muestra las propiedades del descriptor de ordenación en un cuadro de diálogo.

```

Sub DisplaySortDescriptor
  On Error Resume Next
  Dim oSheet
  Dim oRange 'Se necesita un intervalo para crear el descriptor de ordenación.
  Dim oSortDescript
  Dim i%

```

```

Dim s$
Dim oDoc          'Nueva referencia creada al documento de calc.

oDoc = StarDesktop.loadComponentFromURL("private:factory/scalc", "_default",
0, Array())
oSheet = oDoc.Sheets(0)
oRange = oSheet.getCellRangeByName("B28:D33")
oSortDescript = oRange.createSortDescriptor()
For i = LBound(oSortDescript) To UBound(oSortDescript)
    s = s & oSortDescript(i).Name & " = "
    s = s & oSortDescript(i).Value
    s = s & CHR$(10)
Next
MsgBox s, 0, "Descriptor de ordenación"
End Sub

```

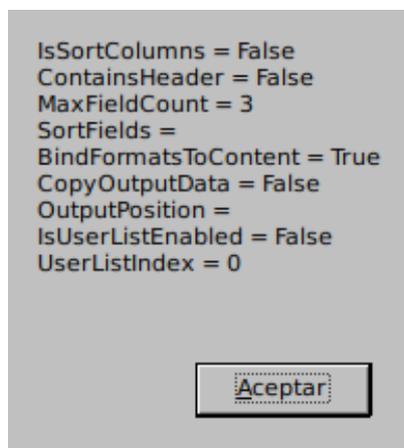


Figura 13: Información de descriptor de ordenación

Filtrado

Un filtro es una herramienta que oculta o muestra registros dentro de una hoja de cálculo, basándose en una serie de criterios de filtrado. De manera similar a la ordenación, los filtros son útiles para reducir grandes listas de datos, con el objetivo de encontrar datos concretos. En Calc existen tres tipos de filtros:

- Filtros automáticos.
- Filtros estándar.
- Filtros avanzados.

Los filtros también se describen en el *Capítulo 2, Introducir, editar y formatear datos*.

Filtro automático

Los filtros automáticos son, de los tres tipos de filtros, los más sencillos, y trabajan introduciendo un cuadro combinado en una o más columnas de datos (*Figura 14*). Para añadir un filtro automático a una o más columnas, simplemente seleccione las columnas, y a continuación seleccione **Datos > Filtro automático** desde la barra de menú. Para acceder al cuadro

combinado del filtro automático de una columna, pulse en el botón de flecha hacia abajo situado en la primera celda de dicha columna.

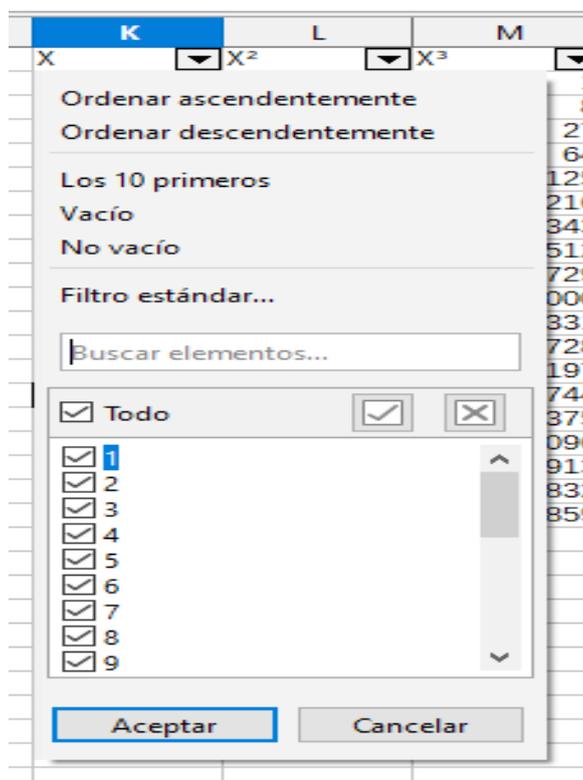


Figura 14: Cuadro combinado de filtro automático

Para eliminar un filtro automático, seleccione de nuevo las columnas y pulse en **Datos > Filtro automático**. Cada cuadro combinado y botón de flecha hacia abajo desaparecerá. En este sentido, la opción del menú actúa como un conmutador para los filtros automáticos.

Cada cuadro combinado de filtro automático tiene los siguientes parámetros de configuración:

- Se puede aplicar una ordenación básica utilizando las opciones *Ordenar ascendentemente* u *Ordenar descendentemente*.
- La opción *Filtro estándar* abre el cuadro de diálogo *Filtro estándar* (Figura 15), y establece automáticamente el campo actual como campo para la primera condición en el cuadro de diálogo.
- Al seleccionar *Vacío*, se ocultan todas las filas no vacías que contienen un valor en la columna actual. Del mismo modo, al seleccionar *No vacío* se ocultan todas las filas vacías que carezcan de un valor en la columna actual. Aquellas filas enteramente vacías, serán ignoradas.
- La selección del filtro *Los 10 primeros*, se mostrarán las diez filas con el mayor valor. Se pueden mostrar más de diez filas, si hay más de diez casos del valor más grande en cada columna. Por ejemplo, si hay once estudiantes con la puntuación perfecta de 100, entonces el filtro mostrará esos once casos.
- Marque la caja *Todo* para mostrar u ocultar todos los valores de la columna actual.
- El filtro automático genera entradas para cada valor único en la columna actual. Estos valores pueden ser filtrados, simplemente marcando cualquiera de las casillas de verificación próximas a cada entrada.

Conmutar Filtros Automáticos con una macro

En Calc, los intervalos de base de datos contienen un valor booleano **AutoFilter** que le permite activar o desactivar los filtros automáticos. La macro del *Listado 4* le muestra cómo realizar esto.

Filtros estándar

Los filtros estándar son más complejos que los filtros automáticos, y permiten hasta ocho condiciones de filtrado. Además, a diferencia de los filtros automáticos, los filtros estándar utilizan un cuadro de diálogo (*Figura 15*), al cual se accede mediante la selección **Datos > Más filtros > Filtro estándar** de la barra de menú.

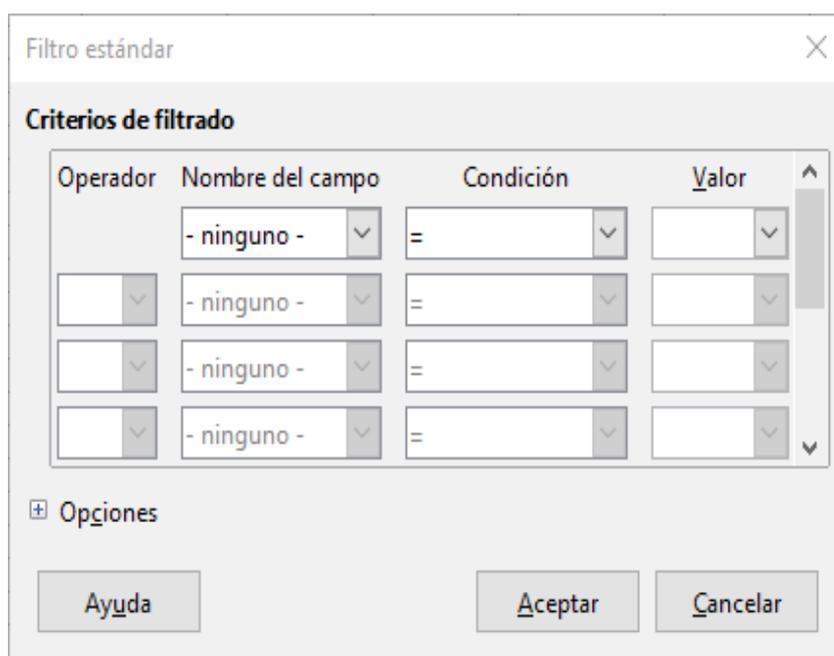


Figura 15: Cuadro de diálogo de Filtro estándar

Para más información sobre cómo utilizar este cuadro de diálogo y sus opciones, vea el *Capítulo 2, Introducir, editar y formatear datos*.

Crear filtros estándar con macros

En una macro, el filtrado de un intervalo de datos se realiza mediante la llamada a la rutina **filter**, bien sobre un intervalo, bien sobre una hoja de cálculo, la cual contenga el intervalo. Como un descriptor de ordenación, un descriptor de filtro contiene la configuración del filtro para la hoja de cálculo actual, como si la primera fila o columna de la hoja contiene encabezados, los cuales no deberán ser filtrados. Un descriptor de filtro se puede generar invocando el método **createFilterDescriptor()** en una hoja de cálculo o sobre una celda del intervalo:

createFilterDescriptor(Empty)

Crea un descriptor de filtro. Si el valor booleano *Empty* se establece a TRUE, entonces se genera un descriptor de filtro vacío. Si *Empty* es FALSE, entonces el descriptor se rellena con la configuración previa del objeto actual (como un intervalo de base de datos).

Al igual que con los descriptors de ordenación, los descriptors de filtros se pueden crear utilizando este método, y luego se pueden modificar y pasar como un argumento al método **filter**. La macro mostrada en el *Listado 8* demuestra este proceso, mediante la creación de un filtro estándar sencillo para la primera hoja de cálculo de un libro. La *Figura 16* muestra los resultados de filtrar la hoja de calificaciones ejemplo de la *Figura 1*.

Listado 8: SimpleSheetFilter crea un filtro estándar sencillo

```
Sub SimpleSheetFilter()  
    Dim oSheet          ' Hoja que contendrá el filtro.  
    Dim oFilterDesc    ' Descriptor del filtro.  
    Dim oFields(0) As New com.sun.star.sheet.TableFilterField  
  
    oSheet = ThisComponent.getSheets().getByIndex(0)  
  
    REM Si el argumento es cierto, crea un descriptor de filtro vacío.  
    REM Si el argumento es falso, crea un descriptor  
    REM con la configuración previa.  
    oFilterDesc = oSheet.createFilterDescriptor(True)  
  
    With oFields(0)  
        REM Puede utilizar la propiedad Connection para indicar  
        REM cómo conectar con el campo previo.  
        REM Este es el primer campo, por lo tanto no es obligatorio.  
        '.Connection = com.sun.star.sheet.FilterConnection.AND  
        '.Connection = com.sun.star.sheet.FilterConnection.OR  
  
        REM La propiedad Field es la columna base número cero a filtrar.  
        REM Si tiene la celda, puede usar  
        REM .Field = oCell.CellAddress.Column.  
        .Field = 5          ' El campo de clasificaciones para Examen #2  
  
        REM Compara usando un valor numérico o una cadena de texto?  
        .IsNumeric = True  
  
        REM La propiedad NumericValue se usa porque anteriormente  
        REM .IsNumeric = True .  
        .NumericValue = 80  
  
        REM Si IsNumeric era False, entonces se usará la propiedad  
        REM StringValue .  
        REM .StringValue = "cualquier cosa"  
  
        REM Operadores válidos incluyen EMPTY, NOT_EMPTY, EQUAL,  
        REM NOT_EQUAL, GREATER, GREATER_EQUAL, LESS,  
        REM LESS_EQUAL, TOP_VALUES, TOP_PERCENT,  
        REM BOTTOM_VALUES, and BOTTOM_PERCENT
```

```

.Operator = com.sun.star.sheet.FilterOperator.GREATER_EQUAL
End With

REM El descriptor de filtro soporta las siguientes
REM propiedades : IsCaseSensitive, SkipDuplicates,
REM UseRegularExpressions,
REM SaveOutputPosition, Orientation, ContainsHeader,
REM CopyOutputData, OutputPosition, y MaxFieldCount.
oFilterDesc.setFilterFields(oFields())
oFilterDesc.ContainsHeader = True
oSheet.filter(oFilterDesc)
End Sub

```

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
3	David	75	86	91	40	88	79	76.5
7	Jennifer	85	93	73	80	100	90	86.83
8	Bethany	95	100	82	80	88	93	89.67
9	Andrew	90	100	82	90	88	95	90.33
10	Ian	100	100	91	90	100	96	96.17

Figura 16: Hoja de calificaciones filtrada por valores mayores de 80 en Examen #2



Nota

Cuando se invoca el método **filter** en una hoja, se oculta cada fila vacía de la hoja. Cuando se invoca **filter** en un intervalo, solo se ocultan las filas vacías del propio intervalo.

Limpiar todos los filtros de una hoja de cálculo

Cuando se aplica un filtro a una hoja de cálculo, se reemplaza cualquier filtro existente en dicha hoja. Por consiguiente, para eliminar un filtro en una hoja, simplemente cree y establezca un filtro vacío para dicha hoja (*Listado 9*).

Listado 9: RemoveSheetFilter elimina el filtro para la hoja actual mediante el establecimiento de un filtro vacío.

```

Sub RemoveSheetFilter()
    Dim oSheet          ' Hoja a filtrar.
    Dim oFilterDesc     ' Descriptor de filtro.

    oSheet = ThisComponent.getSheets().getByIndex(0)
    oFilterDesc = oSheet.createFilterDescriptor(True)
    oSheet.filter(oFilterDesc)
End Sub

```

Filtrar múltiples columnas y filtrar con expresiones regulares

La macro en el *Listado 10* muestra un filtro que filtra dos columnas y utiliza expresiones regulares. Observe que en este ejemplo se invoca el método **filter()** sobre un intervalo, en vez de sobre una hoja de cálculo. La *Figura 17* muestra los resultados de esta macro sobre la hoja ejemplo de calificaciones de la *Figura 1*.

Listado 10: SimpleRangeFilter usa dos columnas

```
Sub SimpleRangeFilter()  
    Dim oSheet          ' Hoja a filtrar.  
    Dim oRange          ' Intervalo a filtrar .  
    Dim oFilterDesc     ' Descriptor de filtro .  
    Dim oFields(1) As New com.sun.star.sheet.TableFilterField  
  
    oSheet = ThisComponent.getSheets().getByIndex(0)  
    oRange = oSheet.getCellRangeByName("A1:H11")  
  
    REM Si el argumento es True, crea un  
    REM descriptor de filtro vacío.  
    oFilterDesc = oRange.createFilterDescriptor(True)  
  
    REM Configura un campo para ver celdas con contenido que  
    REM comiencen con la letra B .  
    With oFields(0)  
        .Field = 0          ' Filtrar columna A (Nombres de estudiantes).  
        .IsNumeric = False  ' Utiliza una cadena, no un número .  
        .StringValue = "b.*" ' Cada nombre comienza con la una B.  
        .Operator = com.sun.star.sheet.FilterOperator.EQUAL  
    End With  
    REM Configura un campo que requiera al menos una de las siguientes condiciones  
    REM Esta nueva condición requiere un valor menor o igual a 90.  
    With oFields(1)  
        .Connection = com.sun.star.sheet.FilterConnection.OR  
        .Field = 6          ' Filtra por columna G (calificaciones Test#1).  
        .IsNumeric = True   ' Usa un número  
        .NumericValue = 90  ' Puntuaciones menores de 90  
        .Operator = com.sun.star.sheet.FilterOperator.LESS_EQUAL  
    End With  
  
    oFilterDesc.setFilterFields(oFields())  
    oFilterDesc.ContainsHeader = True
```

```
oFilterDesc.UseRegularExpressions = True
oRange.filter(oFilterDesc)
End Sub
```

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
5	Bethany	95	100	82	80	88	93	89.67
6	Jennifer	85	93	73	80	100	90	86.83
7	Charles	80	93	73	80	75	84	80.83
8	David	75	86	91	40	88	79	76.5
9	Harley	85	93	82	70	75	76	80.17
10	Ferdinand	85	93	73	60	50	72	72.17
11	Geogia	70	80	55	39	75	67	64.33

Figura 17: Hoja de calificaciones filtrada por puntuaciones de test por debajo del 90%, y nombres de estudiantes que comienzan por "B"

Filtros avanzados

En Calc, los criterios para un filtro avanzado se almacenan en una hoja de cálculo, en vez de introducirse en un cuadro de diálogo. Por consiguiente, primero deberá establecer un intervalo de celdas que contengan los criterios, antes de utilizar el cuadro de diálogo de Filtro avanzado (Figura 18).

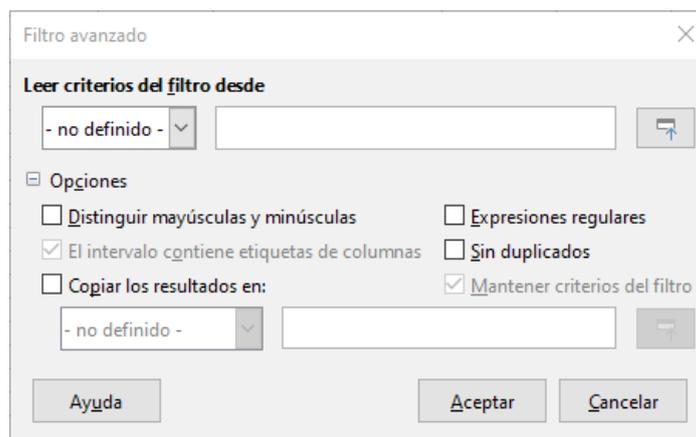


Figura 18: Cuadro de dialogo Filtro avanzado

Para establecer un intervalo de criterios:

- 1) Copie la columna de encabezados del intervalo a ser filtrado a un espacio vacío de una hoja de cálculo. No es necesario que sea en la misma hoja de cálculo en la que se encuentra el intervalo de origen.
- 2) Introduzca los criterios del filtro debajo de los encabezados de las columnas en el intervalo de criterios. Cada criterio individual en la misma fila es conectado con AND, mientras que el grupo de criterios de cada fila está conectado con OR. Se ignoran las celdas vacías. Se podrán definir hasta ocho filas de criterios para un filtro.

Después de crear un intervalo de criterios, configure un filtro avanzado de la siguiente manera:

- 1) Seleccione el intervalo de celdas que desee filtrar.
- 2) Vaya a **Datos > Más filtros > Filtro avanzado** en la barra de menú, para abrir el cuadro de diálogo de Filtro avanzado (Figura 18).

- 3) En el campo *Leer criterios del filtro desde*, introduzca la dirección para el intervalo de nombres, bien mediante la selección del intervalo de nombres desde el desplegable, bien escribiendo una referencia, o seleccionando las celdas de la hoja de cálculo. Recuerde utilizar el botón *Contraer / Expandir* si necesita minimizar temporalmente el cuadro de diálogo durante la selección de las celdas.
- 4) Pulse **Aceptar** para aplicar el filtro y cerrar el cuadro de diálogo.

Las opciones de filtro avanzado son las mismas que las opciones de filtro estándar, y se describen con más detalle en el *Capítulo 2, Introducir, Editar y Formatear Datos*.

Ejemplo de filtro avanzado

La *Figura 19* muestra un ejemplo de filtro de intervalo para la hoja de calificaciones ejemplo de la *Figura 1*.

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
2		>75	>75	>75				
3	Ferdinand							

Figura 19: Intervalo de criterios de filtro avanzado (en hoja 2)

En este intervalo hay dos grupos de criterios: el primero muestra los registros de estudiantes cuyas puntuaciones se encuentra por encima del 75% en cada tarea, y el segundo muestra los registros de cualquier estudiante llamado Ferdinand. La *Figura 20* muestra el resultado de esta operación de filtrado al utilizar estos criterios.

	A	B	C	D	E	F	G	H
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio
2	Ian	100	100	91	90	100	96	96.17
3	Emily	100	100	81	100	75	94	91.67
4	Bethany	95	100	82	80	88	93	89.67
5	Andrew	90	100	82	90	88	95	90.33
7	Harley	85	93	82	70	75	76	80.17
8	Ferdinand	85	93	73	60	50	72	72.17

Figura 20: Hoja ejemplo de clasificaciones filtrada mediante el uso de un filtro avanzado

Usar un filtro avanzado con macros

La aplicación de un filtro avanzado con un macro funciona de manera similar al establecimiento de un filtro estándar. La diferencia principal radica en que la celda del intervalo que contiene el criterio del filtro, se utiliza para crear el descriptor del filtro. La macro en el *Listado 11* muestra cómo se realiza, utilizando para ello el método **createFilterDescriptorByObject**. Utiliza la hoja de calificaciones ejemplo de la *Figura 1* como intervalo de datos y el intervalo de la *Figura 19* como intervalo de criterios. El resultado debería ser el mismo que el mostrado en la *Figura 20*:

Listado 11: AdvancedRangeFilter crea un descriptor de filtro a partir del intervalo de criterios de filtro.

```
Sub AdvancedRangeFilter()
```

```
    Dim oSheet      'Una hoja de un documento de Calc.
```

```
    Dim oRanges     'La propiedad NamedRanges.
```

```
    Dim oCritRange  'Intervalo que contiene el criterio de filtrado.
```

```
    Dim oDataRange  'Intervalo que contiene los datos a filtrar.
```

```

Dim oFiltDesc 'Descriptor de filtrado.

REM Intervalo que contiene el criterio de filtro
oSheet = ThisComponent.getSheets().getByIndex(1)
oCritRange = oSheet.getCellRangeByName("A1:H3")

REM También puede obtener el intervalo que contiene el
REM criterio de filtrado de un intervalo de nombres.
REM oRanges = ThisComponent.NamedRanges
REM oRange = oRanges.getByName("AverageLess80")
REM oCritRange = oRange.getReferredCells()

REM Los datos que desea filtrar
oSheet = ThisComponent.getSheets().getByIndex(0)
oDataRange = oSheet.getCellRangeByName("A1:H11")

oFiltDesc = oCritRange.createFilterDescriptorByObject(oDataRange)
oDataRange.filter(oFiltDesc)
End Sub

```

La *Tabla 3* contiene un listado de propiedades que se corresponden a configuraciones de filtros avanzados (y estándar):

Tabla 3: Propiedades de filtro avanzados y estándar

<i>Función</i>	<i>Descripción</i>
ContainsHeader	Booleano (TRUE/FALSE) que especifica si la primera fila o columna contiene encabezados que no deberían ser filtrados.
CopyOutputData	Booleano que especifica si los datos filtrados deben ser copiados a otra posición en el documento.
IsCaseSensitive	Booleano que especifica si la distinción entre mayúsculas y minúsculas es importante cuando se comparan entradas.
Orientation	Una enumeración que especifica si un intervalo se filtra por columna o por fila: Columna – com.sun.star.table.TableOrientation.COLUMNS Fila – com.sun.star.table.TableOrientation.ROWS
OutputPosition	Si CopyOutputData es TRUE, esta propiedad especifica la posición en la cual serán copiados los datos filtrados.
SaveOutputPosition	Booleano que especifica si la posición OutputPosition es almacenada para futuras llamadas.
SkipDuplicates	Booleano que especifica si entradas duplicadas quedan fuera del resultado.
UseRegularExpressions	Booleano que especifica si las cadenas del filtro son interpretadas como expresiones regulares.

Copiar los resultados de filtros avanzados a ubicaciones diferentes

Los resultados de un filtro avanzado pueden ser extraídos a una ubicación diferente, al utilizar la propiedad **OutputPosition**. Copiar resultados de esta manera, elimina a Calc la necesidad de ocultar las filas que no concuerdan con el criterio de búsqueda, lo que normalmente realizaría si se filtra en la misma ubicación.

El fragmento de código de macro en el *Listado 12* muestra como copiar los resultados de un filtro a una ubicación diferente, y la *Figura 21* muestra los resultados cuando este fragmento se aplica a la macro del *Listado 11*, justo antes de que se invoque el método **filter**. Observe que primero debe modificarse el descriptor del filtro antes de aplicar la configuración de dicho filtro.

Listado 12: Un fragmento de código que copia los resultados del filtro a una ubicación diferente

```
REM Copia los datos de salida en vez de filtrar en el lugar.  
oFiltDesc.CopyOutputData = True  
  
REM Crea una CellAddress y la establece en la Hoja1  
REM Columna B, Fila 16 (recuerde, se comienza a contar en 0)  
Dim outputCell As New com.sun.star.table.CellAddress  
outputCell.Sheet = 0  
outputCell.Column = 1  
outputCell.Row = 15  
oFiltDesc.OutputPosition = outputCell
```



Nota

La propiedad **OutputPosition** devuelve una copia de una estructura. Como resultado, no es posible establecer valores individuales directamente, tales como la fila o columna, mediante esta propiedad. Por ejemplo `oFilterDesc.OutputPosition.Row = 2` no funcionará, ya que es la propiedad *Fila* de la copia la que cambia, no de la original.

	A	B	C	D	E	F	G	H	I
1	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio	
2	Ian	100	100	91	90	100	96	96.17	
3	Emily	100	100	81	100	75	94	91.67	
4	Bethany	95	100	82	80	88	93	89.67	
5	Andrew	90	100	82	90	88	95	90.33	
6	Ferdinand	85	93	73	60	50	72	72.17	
7	Harley	85	93	82	70	75	76	80.17	
8	Jennifer	85	93	73	80	100	90	86.83	
9	Charles	80	93	73	80	75	84	80.83	
10	David	75	86	91	40	88	79	76.5	
11	Geogia	70	80	55	39	75	67	64.33	
12									
13									
14									
15									
16	Alumno	HW #1	HW #2	HW #3	Examen 1	Examen 2	Prueba 1	Promedio	
17	Ian	100	100	91	90	100	96	96.17	
18	Emily	100	100	81	100	75	94	91.67	
19	Bethany	95	100	82	80	88	93	89.67	
20	Andrew	90	100	82	90	88	95	90.33	
21	Harley	85	93	82	70	75	76	80.17	
22									
23									

Figura 21: Resultados de Filtro avanzado copiados a la celda B16

Funciones útiles similares a bases de datos

Calc tiene muchas funciones que son utilizadas frecuentemente en el contexto de las bases de datos. Algunas de estas funciones son fáciles de utilizar (como SUM) o resultan familiares en el contexto en el cual son típicamente usadas (como STDEV para estadísticas). Unas pocas, como las funciones LOOKUP, son un tanto más infrecuentes, pero sin embargo conocerlas es útil si piensa utilizar Calc para tablas de bases de datos. Esta sección proporciona un listado condensado de estas funciones. Puede encontrar más material de referencia en el *Capítulo 18, Descripción de funciones*, así como en la ayuda en línea, en la dirección <http://help.libreoffice.org/>.



Nota

Las funciones con el sufijo -A, tratan valores de texto como un número, con el valor de cero. Estas funciones ignoran las celdas en blanco.

Función	Categoría	Descripción
PROMEDIO	Estadística	Devuelve la media de sus argumentos. Ignora celdas vacías y celdas que contienen texto.
PROMEDIOA	Estadística	Devuelve la media de sus argumentos, pero solo ignora celdas vacías. El valor del texto es 0.
PROMEDIO.SI	Estadística	Devuelve la media aritmética de todas las celdas en un intervalo que satisface la condición dada.
PROMEDIO.SI.CONJUNTO	Estadística	Devuelve la media aritmética de todas las celdas en un intervalo que satisface múltiples criterios dados.
CONTAR	Estadística	Cuenta el número de valores numéricos en una lista de argumentos. Ignora las celdas vacías y las celdas que contienen texto.

Función	Categoría	Descripción
CONTARA	Estadística	Cuenta el número de valores en una lista de argumentos, pero cuenta tanto argumentos numéricos como de texto. Se ignoran celdas vacías.
CONTAR.BLANCO	Estadística	Devuelve el número de celdas vacías dentro de un intervalo.
CONTAR.SI	Estadística	Devuelve el número de celdas en un intervalo que cumplen el criterio de búsqueda especificado.
CONTAR.SI.CONJUNTO	Estadística	Devuelve el número de celdas que cumplen los criterios en múltiples intervalos.
BUSCARH	Hoja de cálculo	Busca un valor en la primera fila de una matriz y devuelve un valor de una fila diferente en la misma columna.
INDICE	Hoja de cálculo	Devuelve los contenidos de una celda en un índice especificado (denotado por los números de fila y columna) dentro de un intervalo.
INDIRECTO	Hoja de cálculo	Devuelve la referencia especificada por una cadena de texto.
BUSCAR	Hoja de cálculo	Devuelve los contenidos de una celda, contenida dentro de una única fila o columna, de un intervalo o de una matriz.
COINCIDIR	Hoja de cálculo	Busca un elemento dentro de una matriz, y devuelve su posición relativa dentro de la matriz.
MAX	Estadística	Devuelve el valor más grande en una lista de argumentos.
MAXA	Estadística	Devuelve el valor más grande en una lista de argumentos. El valor de un texto es 0.
MAX.SI.CONJUNTO	Estadística	Devuelve el valor más grande en las celdas de un intervalo, que cumple múltiples criterios en múltiples intervalos.
MEDIANA	Estadística	Devuelve la mediana de una lista de números.
MIN	Estadística	Devuelve el valor más pequeño de una lista de argumentos.
MINA	Estadística	Devuelve el valor más pequeño de una lista de argumentos. El valor de un texto es 0.
MIN.SI.CONJUNTO	Estadística	Devuelve el valor más pequeño de las celdas de un intervalo, que cumple múltiples criterios en múltiples intervalos.
MODO	Estadística	Devuelve el valor más común en una lista de valores.
DESREF	Hoja de cálculo	Devuelve el valor de una celda desplazada un cierto número de filas y columnas, desde un punto de referencia dado.
PRODUCTO	Matemática	Multiplica todos los números en una lista de argumentos, y devuelve el producto.
DESVEST DESVEST.M	Estadística	Calcula la desviación estándar de una muestra de población.
DESVESTA	Estadística	Calcula la desviación estándar de una muestra de población. El valor de un texto es 0.

Función	Categoría	Descripción
DESVESTP	Estadística	Calcula la desviación estándar de una población completa.
DESVESTPA	Estadística	Calcula la desviación estándar de una población completa. El valor de un texto es 0.
SUBTOTALES	Matemática	Calcula el total de un subconjunto de una matriz, que es creado utilizando filtros automáticos.
SUMA	Matemática	Devuelve la suma de una lista de valores.
SUMAR.SI	Matemática	Calcula la suma de valores de celdas que cumplen el criterio de búsqueda especificado.
SUMAR.SI.CONJUNTO	Matemática	Devuelve la suma de valores de celdas en un intervalo, que cumplen múltiples criterios en múltiples intervalos.
VAR VAR.S	Estadística	Calcula la varianza de una muestra de población.
VARA	Estadística	Calcula la varianza de una muestra de población. El valor de un texto es 0.
VARP VAR.P	Estadística	Calcula la varianza de una población completa.
VARPA	Estadística	Calcula la varianza de una población completa. El valor de un texto es 0.
BUSCARV	Hoja de cálculo	Busca un valor en la primera columna de una matriz y devuelve un valor de una columna diferente en la misma fila.

Funciones específicas de base de datos

Algunas funciones de Calc están específicamente diseñadas para usarse con una tabla de base de datos. Con una excepción (DGET), estas funciones son formas especializadas de funciones comúnmente usadas, tales como COUNT, y todas ellas son denotadas con el prefijo D- (como DAVERAGE). En la *Tabla 4* se muestra una breve lista de estas funciones, mientras que en el *Capítulo 18, Descripción de Funciones*, se presentan descripciones más detalladas.



Nota

La *Tabla 4* utiliza los siguientes términos de manera intercambiable: fila y registro; columna y campo.

Tabla 4: Funciones de base de datos en un documento de Calc

Función	Descripción
BDPROMEDIO	Devuelve la media de todos los campos que cumplen con el criterio de búsqueda.
BDCONTAR	Cuenta el número de registros que contienen datos numéricos que cumplen con el criterio de búsqueda.
BDCONTARA	Cuenta el número de registros que contienen datos numéricos o alfanuméricos, que cumplen con el criterio de búsqueda.
BDEXTRAER	Devuelve los contenidos de un campo que cumplen el criterio de búsqueda especificado.

Función	Descripción
BDMAX	Devuelve el valor máximo de un campo para cada registro, que cumplen el criterio de búsqueda especificado.
BDMIN	Devuelve el valor mínimo de un campo para cada registro, que cumplen el criterio de búsqueda especificado.
BDPRODUCTO	Devuelve el producto de todos los valores de un campo, que cumplen el criterio de búsqueda.
BDESVEST	Calcula la desviación estándar de todos los valores de un campo, que cumplen con el criterio de búsqueda. Los valores son tratados como una muestra.
BDESVESTP	Calcula la desviación estándar de todos los valores de un campo, que cumplen con el criterio de búsqueda. Los valores son tratados como la población completa.
BDSUMA	Suma todos los valores de un campo que cumplen con el criterio de búsqueda.
BDVAR	Calcula la varianza de todos los valores de un campo, que cumplen el criterio de búsqueda. Los valores son tratados como una muestra.
BDVARP	Calcula la varianza de todos los valores de un campo que cumplen el criterio de búsqueda. Los valores son tratados como una población completa.