

CONTROLES, MACROS Y FUNCIONES

CONTENIDO

Controles, macros y funciones	1
Contenido	1
Macros	2
Ideas previas	2
Seguridad de las macros	5
Grabación de macros	6
Asignación de una macro a un objeto	9
Uso de controles	15
Botones	15
Otros controles	21
Barra de desplazamiento	25
Macros de apertura	27
Definición de funciones	29

MACROS

IDEAS PREVIAS

¿Qué es una macro?

Una macro es un conjunto de operaciones de Hoja de Cálculo que el programa memoriza para repetir ese conjunto cada vez que se desee.

En el uso de macros podemos distinguir tres operaciones:

- Usar macros que han programado otras personas.
- Grabar y ejecutar nuestras propias macros
- Aprender a programar macros mediante el lenguaje Basic de LibreOffice

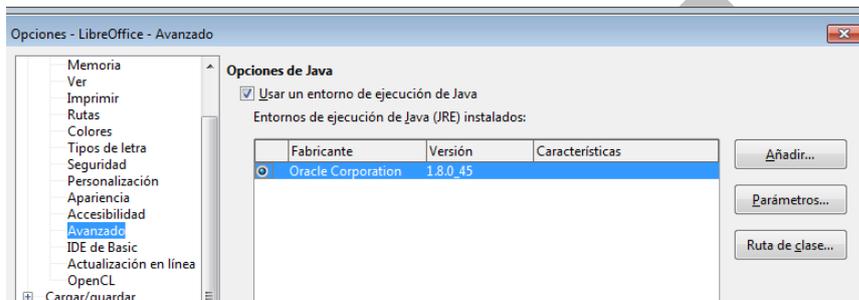
En esta guía estudiaremos los dos primeros usos y algo del tercero.

Operación previa

En las versiones más recientes de Calc has de tener implementada una máquina Java para poder usar macros. Lo normal es que dispongas ya de Java, porque es útil en otras tareas. Si no es así, descárgate esta herramienta. Basta que escribas “instalar Java” en un buscador y obtendrás un sitio de descarga como este



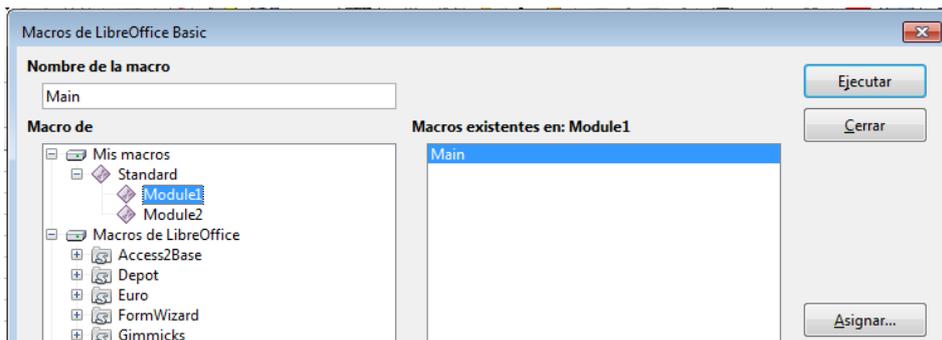
Una vez dispongas del entorno Java, has de comunicárselo a Calc. Para ello usa **Herramientas – Opciones** y busca a la izquierda la palabra **Java**. Pincha sobre ella, activa la opción de *Usar un entorno de ejecución Java instalado* y espera unos segundos hasta que aparezcan a la derecha los entornos Java de tu equipo.



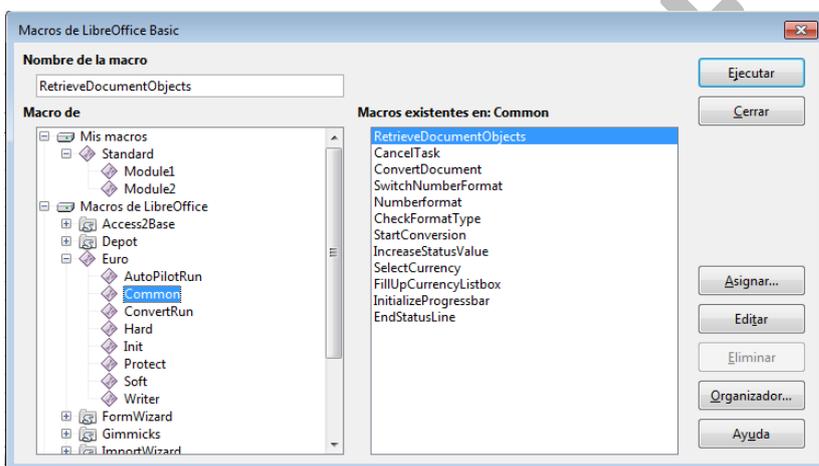
Elige uno y pulsa en **Aceptar**. Con eso ya puedes grabar y crear macros.

¿Dónde se guardan las macros?

Las macros están alojadas en unos contenedores llamados *módulos*. Pueden pertenecer a las carpetas **Mis macros** o a **Macros de LibreOffice**, en cuyo caso se pueden ejecutar en cualquier hoja de cálculo, o bien estar destinadas sólo a una hoja, y estas serán las que nos interesen en este momento, porque las otras es peligroso tocarlas.



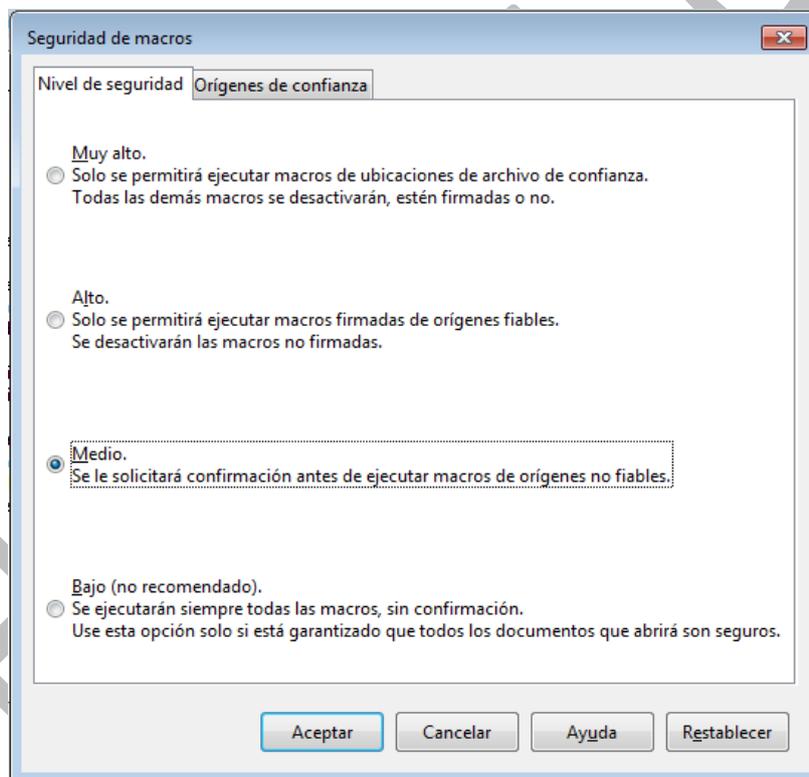
Usa la secuencia **Herramientas - Macros - Organizar macros - LibreOffice Basic** para ver qué macros tiene definidas tu hoja de cálculo. En la imagen se pueden ver algunas de las pertenecientes a Macros de LibreOffice



Estudia bien esta ventana. A la izquierda figuran las carpetas **Mis macros** y **Macros de LibreOffice**, y en esta hemos abierto el módulo **Euro** y la subcarpeta **Common**. A la derecha figuran las macros que contiene este módulo. No toques nada, que puedes alterar el funcionamiento de tu hoja.

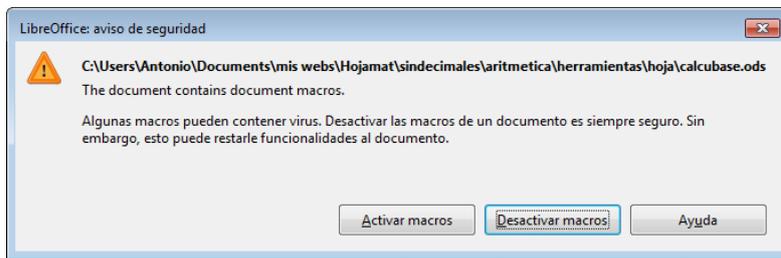
SEGURIDAD DE LAS MACROS

Como una macro puede contener virus, aunque últimamente no suelen aparecer, LibreOffice define cuatro niveles de seguridad para el uso de las mismas. Puedes cambiar el nivel con la secuencia **Herramientas - Opciones... - LibreOffice - Seguridad**, pulsando después sobre el botón **Seguridad de macros...** Se te ofrecerán los niveles **Muy alto, Alto, Medio y Bajo**.



No se recomienda el **Bajo**, porque equivale a dejar una puerta abierta a posibles virus. Si sólo vas a usar macros de fuentes de confianza, puedes elegir el nivel **Medio**, que te avisa de la existencia de macros al abrir una hoja, para que decidas si se desactivan o no. Ten en cuenta que no

funcionarán si las desactivas, con lo que la hoja de cálculo puede perder su sentido.



GRABACIÓN DE MACROS

Para que te inicies en la grabación de macros, crearemos una que borre un rango de celdas, por ejemplo B4:D12

Para crear una macro mediante grabación has de seguir esta secuencia:

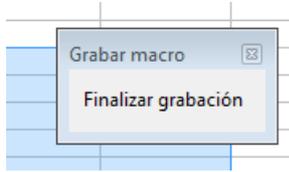
1. Pides **Herramientas - Macros - Grabar macro**
2. Realizas manualmente las operaciones que deseas que automatice la macro, en este caso borrar el rango B4:D12
3. Decides **Finalizar la grabación**
4. **Guardas la macro** en un módulo contenedor.
5. **Ejecutas la macro** para comprobar que la has grabado bien.

Intenta recorrer esa secuencia con el ejemplo propuesto:

1. Recuerda bien qué celdas deseas borrar. En este caso, **desde B4 hasta D12**

Activa la secuencia de menú **Herramientas - Macros - Grabar Macro**.

Te aparecerá una pequeña ventana con el botón de **Finalizar Grabación**. Ignórala por ahora. Déjala abierta sin tocarla.



2. Realiza manualmente las operaciones para borrar los datos:

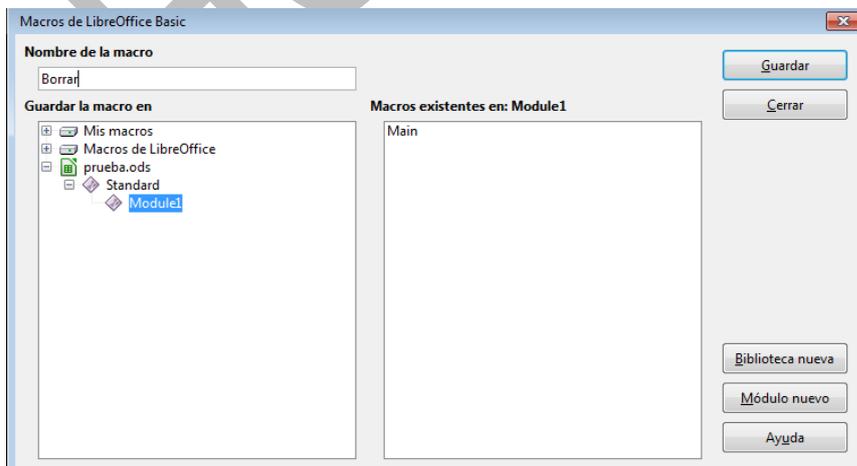
- (a) Selecciona los datos B4:D12
- (b) Pulsa la tecla Supr para borrar contenidos

3. Pulsa sobre el botón de **Finalizar grabación**

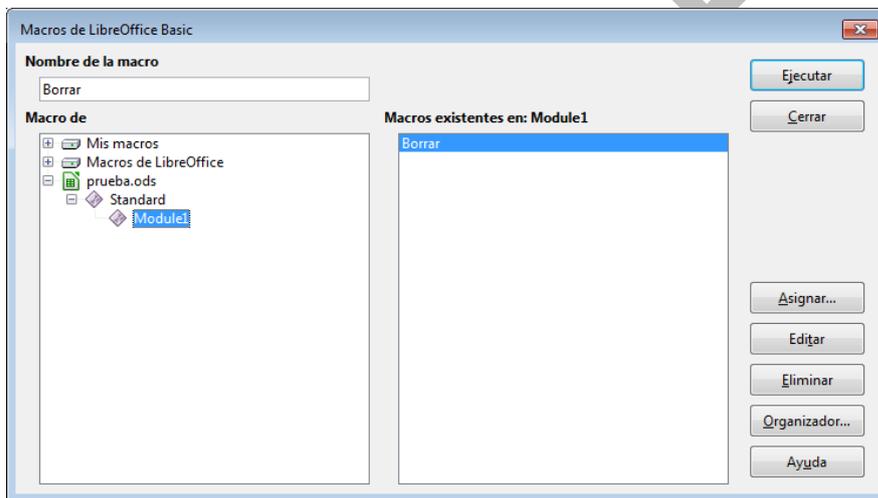
Con esto ya tienes grabada la macro. Pasamos a guardarla. Imagina que tu hoja se llama prueba.odf.

4. Guardar macro

En cuanto des al botón de Finalizar se te abrirá esta ventana.



Pulsa sobre el signo + de la carpeta prueba.ods para que se abra la subcarpeta Standard, como ves en la imagen. Ahora debes decidir en qué módulo (contenedor de macros) vas a guardar tu macro y cómo la vas a llamar. Para ello pulsa sobre el botón **Nuevo módulo** y se te ofrecerá un módulo llamado Module1 o Module2 o similar. Puedes conservar el nombre o cambiarlo. Supongamos que se llama Module1. Pulsa el botón de Aceptar y observarás que se ha creado una subcarpeta de Standard llamada Module1. El programa ha creado una macro llamada Main que puedes ignorar.



Para darle un nombre a la macro que has grabado, en la línea de **Nombre de la macro** escribe, por ejemplo, **Borrar** y pulsa sobre el botón **Guardar**. Se cerrará la ventana, dejando guardada macro con el nombre de Borrar.

5. Ejecutar la macro Borrar.

Escribe varios números en el rango B4:D12, para probar tu macro.

Pide ahora **Herramientas - Macros... - Ejecutar Macro**

Busca tu macro abriendo sucesivamente las carpetas **prueba.ods - Standard - Module1** (u otro nombre) y la verás. Este proceso puede tardar algo.

Ya solo tienes que seleccionarla y pulsar sobre el botón **Ejecutar**. Si has seguido las instrucciones correctamente, se deberán borrar todos los datos.

Ahora puedes repetir la operación cuantas veces desees: escribe datos y ejecuta la macro, y se borrarán en un instante.

ASIGNACIÓN DE UNA MACRO A UN OBJETO

La forma de ejecutar una macro mediante la secuencia **Herramientas - Macros... - Ejecutar macro** es muy lenta. Sería preferible ejecutar macros con una sola pulsación de ratón. Para ello basta con asignar la macro a un objeto, como una imagen, autoforma, rótulo fontwork, cuadro de texto, botón etc.

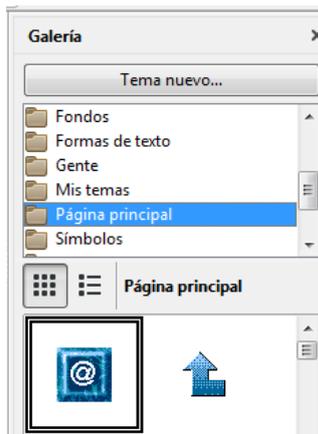
Lo veremos con varios ejemplos, y simultáneamente aprenderemos algunas operaciones sobre celdas.

Incremento del valor de una celda

Escribe un número cualquiera en la celda B4 y algún rótulo de presentación sobre ella. Por ejemplo así:

	A	B	C	
1				
2	Incremento de una celda			
3				
4		56		
5				

Ahora, con **Ver - Galería - Página principal**, arrastra una imagen de botón junto a la celda B4.



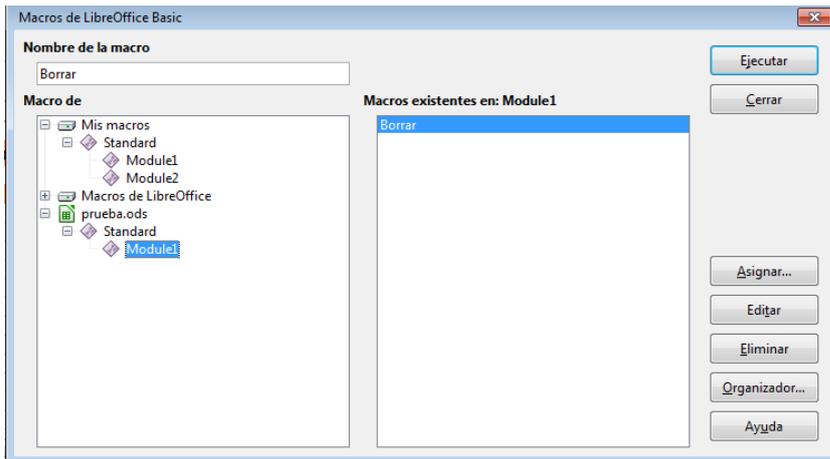
También puedes acceder desde el Panel Lateral

Si no dominas esta operación, inserta una imagen cualquiera.

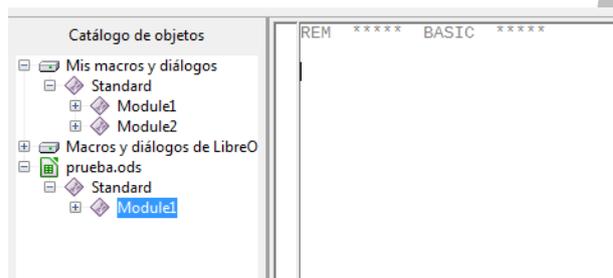
	A	B	C	D
1				
2	Incremento de una celda			
3				
4		56		
5				

Guarda la hoja con el nombre ***prueba.ods*** u otro similar.

Accede a la edición de una macro nueva siguiendo los pasos **Herramientas - Macros - Organizar macros - LibreOffice Basic...** Si no tienes módulo contenedor, créalo como ya vimos en el apartado anterior.



Si no se abre de forma automática el editor de macros, pulsa el botón **Editar** y obtendrás una pantalla de edición, en la que quizás esté iniciada la macro **Main**, que puedes borrar o ignorar.



Escribe debajo este código de macros (puedes copiarlo desde esta misma página web con CTRL-C y CTRL-V), aunque todavía no lo entiendas:

Sub incrementar

v=StarDesktop.CurrentComponent.sheets(0).GetCellByPosition(1,3).value

v=v+1

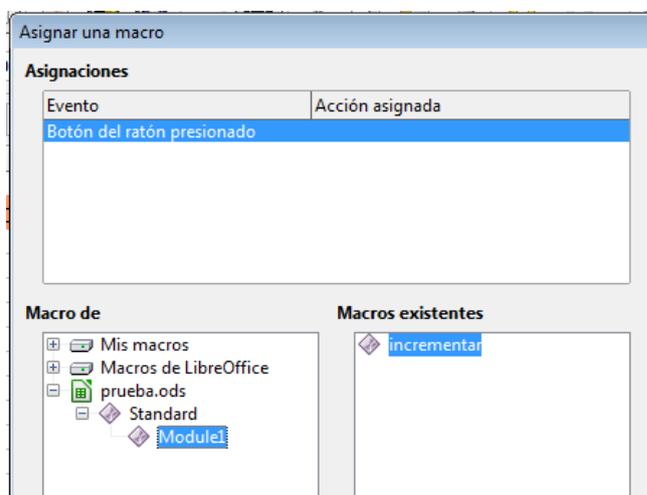
StarDesktop.CurrentComponent.sheets(0).GetCellByPosition(1,3).value=v

end sub



Esta macro te incrementa de 1 en 1 el valor de la celda B4. Después la explicaremos.

Ahora viene lo más fácil: Señala la imagen del botón y pulsa el botón derecho. Elige la opción de **Asignar macro...** Se te abrirá este cuadro de diálogo (a veces tarda mucho en aparecer la lista de macros)



Busca tu archivo prueba.ods, el módulo Module1 (u otro que tengas) y la macro **incrementar**. Selecciónala y pulsa **Asignar** y después **Aceptar**.

Con esto, cada vez que pulses sobre la imagen (el puntero del ratón se transformará en un dedo señalador) se incrementará el valor de la celda B4. Si no te funciona será porque los objetos estén en modo diseño. Si no tienes a la vista la barra de herramientas **Controles de formulario**, ábrela con **Ver – Barras de herramientas**. En esta barra

dispones del botón para cambiar el modo diseño a modo ejecución:



Púlsalo, y si el resto de objetos de la barra los ves agrisados, es que ya estás en modo ejecución. Entonces te funcionará el botón que has creado.

Explicación del código

Aunque este documento no pretende enseñarte a programar, te vendrá bien saber interpretar el código que has copiado:

Sub incrementar

```
v=StarDesktop.CurrentComponent.sheets(0).GetCellBy  
Position(1,3).value  
v=v+1  
StarDesktop.CurrentComponent.sheets(0).GetCellByPo  
sition(1,3).value=v  
end sub
```

Lo explicamos por partes:

Todas las macros han de comenzar con la palabra clave **Sub** seguida del nombre de la macro (en este caso **incrementar**) y han de terminar con la línea **end sub**.

En la segunda línea y en la cuarta se usan tres palabras que conviene interpretar bien. Son nombres de objetos de LibreOffice.org.

StarDesktop.CurrentComponent: Se refiere al archivo actual de LibreOffice.org que tienes abierto.

sheets(0): Se refiere a la hoja en la que se actuará. Se usa el cero para referirse a la primera hoja, porque la numeración en los objetos comienza siempre con el 0 y no con el 1.

GetCellByPosition(1,3): Es la referencia al objeto celda B4. El 1 se refiere a la columna B, que es la segunda, pero como comenzamos a contar por el cero, hay que escribir un 1. El 3 se refiere a la cuarta fila, que también hay que situar contando desde cero.

Explicado esto, se entiende mejor lo que ejecuta la macro:

Línea 1: Comienzo obligatorio con la palabra Sub y el nombre de la macro.

Línea 2: Se lee el valor de la celda B4 (palabra value) y el resultado se guarda en una variable llamada v.

Línea 3: Se incrementa v en una unidad.

Línea 4: Se copia el valor de v en la celda B4 de nuevo, pero ya incrementado.

Línea 5: Sentencia obligatoria end sub

La explicación anterior sólo pretende que te vayas acostumbrando a la forma de trabajar en Basic. Un aprendizaje más profundo requiere un manual sistematizado.

USO DE CONTROLES

BOTONES

Se llama control a cualquier objeto activo, con sus propiedades y métodos, que se inserte en la hoja de cálculo para actuar de interfase entre los usuarios y la hoja. Son controles los botones de acción, las casillas de verificación, las listas desplegables, etc. Son elementos que has visto en cualquier programa de ordenador, que también se pueden usar en LibreOffice.org.

Para ver cuántos controles puedes usar, pide **Ver - Barras de herramientas** y activa la barra de **Campos de control de formulario**.



Recorre uno a uno con el ratón sin pulsar para aprenderte sus nombres.

Un caso práctico

Diseñaremos un botón tal que al pulsar sobre él aparezca una ventana nueva en la que figure escrita la fecha actual. Los pasos que hemos de dar son:

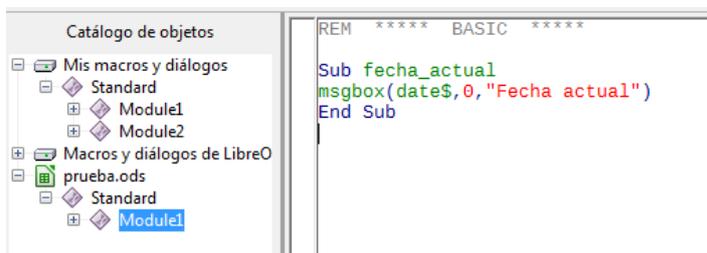
Preparar una macro

Repite los pasos que diste para definir la macro del apartado anterior, la que incrementaba el valor de la celda

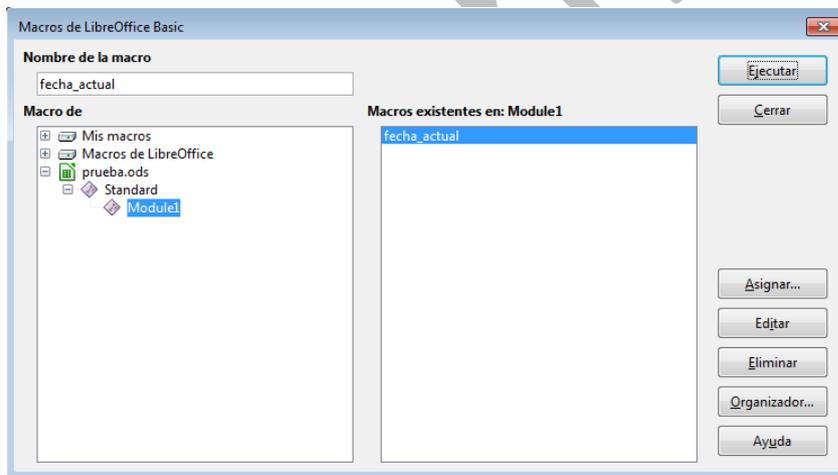
B4: crea un archivo de hoja de cálculo llamado, por ejemplo, fecha.ods.

Accede al editor de macros como ya has aprendido, pero ahora escribe lo siguiente:

```
Sub fecha_actual  
msgbox(date$,0,"Fecha actual")  
End Sub
```



Guarda el archivo que has creado.



Diseñar un botón

Asegúrate de tener a la vista la barra de controles:

Si no es así, actívala con el menú Ver, como ya se indicó antes.



Si los controles se ven en gris desvaído, es que no está activo **el modo de diseño**. Para activarlo pulsa el segundo botón, el que contiene la expresión “OK”. 

Pincha sobre el icono  de botones

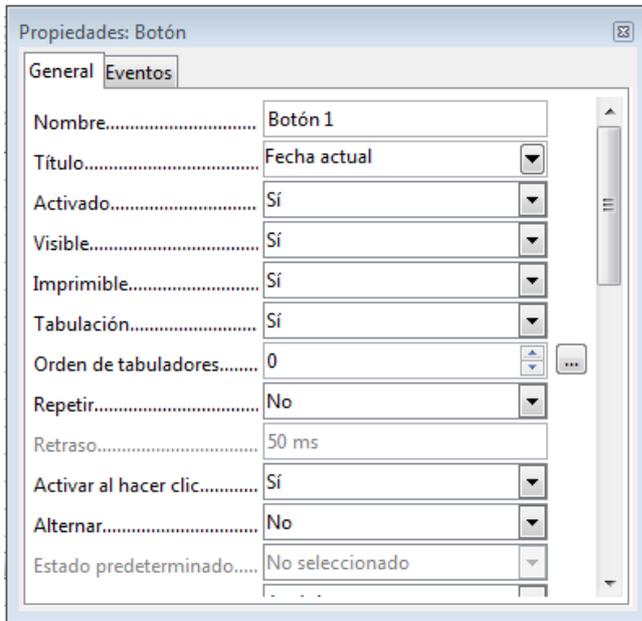
Pasa a cualquier celda de la hoja y, sin soltar el dedo del ratón, dibuja un rectángulo con el mismo:



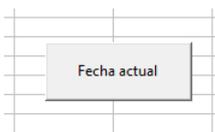
Ya tienes creado un botón de acción.

Sería conveniente cambiarle el nombre, por ejemplo a **Fecha actual**. Para ello, pulsa sobre él con el botón derecho y elige **Control...**

Obtendrás una completa ventana de propiedades. Busca la pestaña **General** y en ella la propiedad **Título**. Cámbialo a "Fecha actual".



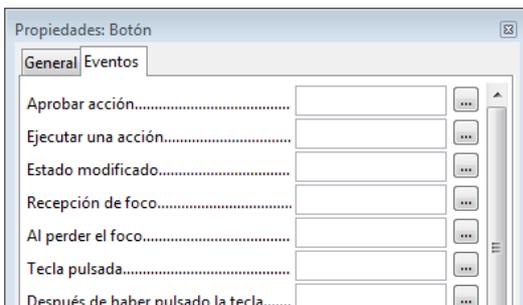
De forma inmediata cambiará su título.



Asignación de la macro

Para que el botón sea útil deberemos asignarle una macro que se ejecute al pulsarlo. Los pasos a dar son los siguientes:

Pulsa con el botón derecho del ratón sobre el botón. Elige **Control...** En la ventana que se abre elige la pestaña **Eventos** (asegúrate de estar en Modo Diseño).

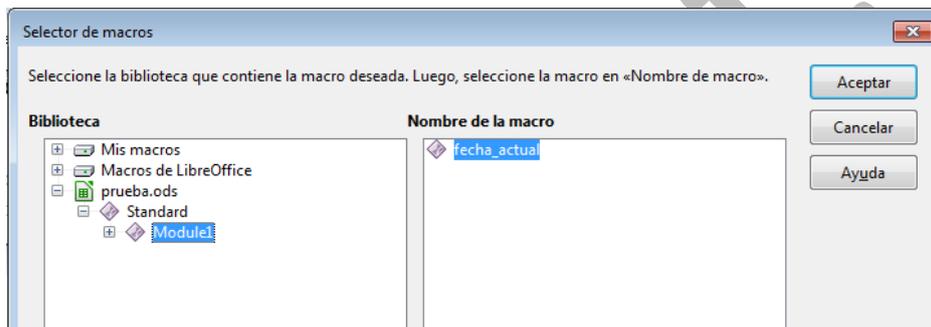


Pulsa ahora sobre el botón de puntos suspensivos situado a la derecha de la frase "**Ejecutar una acción**"

En la siguiente ventana elige **Ejecutar un acción** y pulsa sobre el botón Macro...

Al pulsar puede ser que tarde un poco en aparecer la ventana de asignación. Espera sin inquietarte.

Busca la macro **fecha_actual**, selecciónala y pulsa **Aceptar**.



Con estas operaciones has terminado de configurar el botón de acción. Ahora debe funcionar. Pulsa **Aceptar** en todas las ventanas y cierra la de propiedades.

Ejecución desde el botón

En primer lugar, asegúrate **de que has salido del modo de diseño** en la barra de controles. Si no tienes la seguridad, pulsa varias veces en el botón  hasta que veas los controles en gris desvaído.

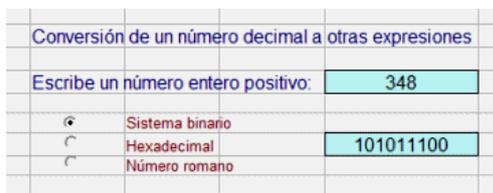
Eso será señal de que has entrado en el modo de ejecución.

OTROS CONTROLES

Además de los botones, puedes usar todo un catálogo de controles. No vamos a estudiarlos todos, pero incluiremos dos tipos para que te animes a investigar por tu cuenta.

BOTONES DE OPCIÓN

Estos botones, casi siempre agrupados en conjuntos, nos sirven para elegir entre varias posibilidades. Para conocerlos mejor nos plantearemos un conversor de un número decimal a las expresiones binaria, hexadecimal y romana.



En la imagen puedes observar el grupo de botones de opción que servirá para elegir la expresión deseada. Según se pulse sobre uno u otro, aparecerá una expresión distinta.

Para construir este modelo de cálculo deberemos seguir estos pasos:

Construcción de los botones

Busca la Barra de controles. Si no la encuentras acude al comando **Ver**.



Habilita el modo **Diseño** pulsando sobre el botón 

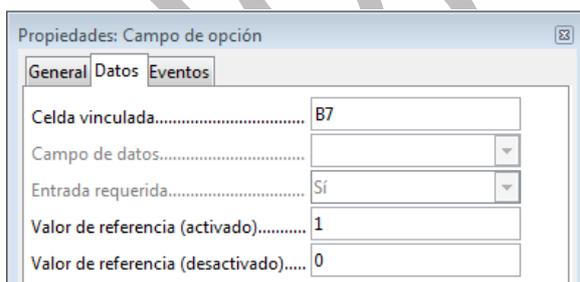
Pulsa sobre el botón  de opción

Sin soltar el dedo del botón izquierdo del ratón, señala la celda B7 y abre sobre ella un rectángulo que acogerá el botón de opción. Haz lo mismo sobre la celda B8 y sobre la B9. Deberás tener tres botones de opción en columna. Si no están bien alineados, mueve algunos de ellos arrastrando con el ratón.



Señala los botones uno a uno, pulsa el botón derecho, elige **Control...** y abre la pestaña **General** de la ventana de diálogo que se abre. Borra el título y déjalo en blanco para que no estorbe. Haz lo mismo con los tres botones.

Pasa ahora a la pestaña de **Datos** y rellénalos para cada botón. Para el primero fija la celda vinculada la B7, como Valor activado 1 y como desactivado 0. Los tienes en la siguiente imagen:



Para el segundo botón fija la celda B8 y los mismos valores 1 y 0. Igual para el tercero, pero con la celda B9.

El sentido de todo esto es que si se pulsa el primer botón, la celda B7 valdrá "1" y en caso contrario, "0". La celda B8

valdrá "1" cuando se pulse el segundo botón, e igual ocurrirá con la B9 y el tercero.

El problema es que los valores se ven, y no queda estético

1	<input checked="" type="radio"/>	Sistema binario
0	<input type="radio"/>	Hexadecimal
0	<input type="radio"/>	Número romano

Para que no se vean, asigna a las celdas B7, B8 y B9 el color de texto blanco.

Escribe los demás textos que ves en la imagen

Conversión de un número decimal a otras expresiones		
Escribe un número entero		348
<input checked="" type="radio"/>	Sistema binario	
<input type="radio"/>	Hexadecimal	101011100
<input type="radio"/>	Número romano	

El número escrito en el sistema decimal lo escribes en la celda E5. Finalmente, en la celda E8 ha de incluirse la siguiente fórmula:

=SI(B7="1";DEC.A.BIN(E5);SI(B8="1";DEC.A.HEX(E5);ROMANO(E5)))

que decide según el botón que esté activado: Si es el primero, se usa la función de conversión DEC.A.BIN(E5), si es el segundo, DEC.A.HEX(E5) y si es el tercero, ROMANO(E5). Aquí tienes algunos ejemplos:

Conversión de un número decimal a otras expresiones		
Escribe un número entero		2015
<input type="radio"/>	Sistema binario	
<input type="radio"/>	Hexadecimal	MMXV
<input checked="" type="radio"/>	Número romano	

Conversión de un número decimal a otras expresiones		
Escribe un número entero		455
<input type="radio"/>	Sistema binario	
<input checked="" type="radio"/>	Hexadecimal	1C7
<input type="radio"/>	Número romano	

CASILLAS DE VERIFICACIÓN

Las casillas de verificación se diferencian de los anteriores controles en que no son exclusivas. Se pueden elegir varias a la vez. Para explicarlas propondremos la construcción de una suma de varios datos en los que se podrá elegir qué datos se suman dentro del conjunto.

	Sumas parciales
<input type="checkbox"/>	7823
<input checked="" type="checkbox"/>	29981
<input checked="" type="checkbox"/>	3499
<input type="checkbox"/>	21087
<input checked="" type="checkbox"/>	4463
Total	37943

Crea un modelo de suma como el de la imagen: Las casillas de verificación las hemos situado en el rango C3:C7 y los datos en D3:D7.

Los pasos a dar son los mismos que en el anterior control:

- Activar Modo Diseño
- Crear las cinco casillas
- Para cada una borramos el título, asignamos una de las celdas desde C3 hasta C7, con valor 1 el activado y 0 el desactivado.
- A las celdas C3:C7 las dotamos de color de texto blanco para que no se vean los valores 0 y 1.

Para el total usamos la fórmula

=C3*D3+C4*D4+C5*D5+C6*D6+C7*D7

-
- Desactivamos el Modo Diseño

Si marcamos sólo dos o tres casillas, la suma sólo abarcará los datos señalados. Insertamos alguna otra suma:

	Sumas parciales
<input type="checkbox"/>	7823
<input type="checkbox"/>	29981
<input checked="" type="checkbox"/>	3499
<input checked="" type="checkbox"/>	21087
<input checked="" type="checkbox"/>	4463
Total	29049

	Sumas parciales
<input checked="" type="checkbox"/>	7823
<input type="checkbox"/>	29981
<input type="checkbox"/>	3499
<input type="checkbox"/>	21087
<input checked="" type="checkbox"/>	4463
Total	12286

BARRA DE DESPLAZAMIENTO

Lo que sigue es una curiosidad, pero que puede ayudar en cuestiones del tipo “¿Qué pasaría si...? Imaginemos que una variable puede presentar valores entre dos extremos. Con una barra de desplazamiento puedes recorrer esos valores de una forma rápida y ver los resultados mientras mueves el control.

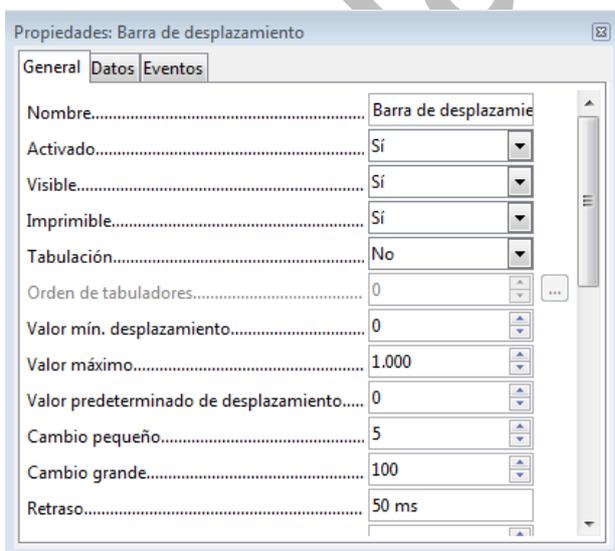
Lo vemos con un ejemplo muy conocido:

Con 10 metros de cerca deseamos construir un espacio rectangular para cuidar unas gallinas. La cerca estará

adosada a una pared, por lo que sólo necesitamos contar con la base del rectángulo y los dos lados laterales. Su perímetro será $P=a+b+b=a+2b$, y la superficie $S=a*b$. Como sabemos que P son 10 metros o 1000 cm., el valor de b se puede expresar como $(10-a)/2$ y la superficie como $S=a*(10-a)/2$. Podemos plantear que a varíe entre 0 y 1000 cm. y tomar nota de la evolución de la superficie. Lo hemos implementado así:

Longitud cerca	1000
Base del rectángulo	400
El otro lado	300
Superficie	120000

Hemos programado una barra de desplazamiento (búscala en la Barra de controles) y la hemos ajustado para que su valor se refleje en la celda correspondiente a la base del rectángulo. Aquí tienes los parámetros:



Los valores irán de 0 a 1000, con cambios pequeños de 5 unidades y cambios grandes (cuando pulsas con el ratón en los huecos de la barra de desplazamiento) de 100. Después se han vinculado sus valores a la celda E5, que es la que contiene la base del rectángulo. Aquí tienes varias imágenes de cómo el desplazamiento se traduce a valores de la base:

Longitud cerca	1000
Base del rectángulo	0
El otro lado	500
Superficie	0

Longitud cerca	1000
Base del rectángulo	288
El otro lado	356
Superficie	102528

Longitud cerca	1000
Base del rectángulo	771
El otro lado	114,5
Superficie	88279,5

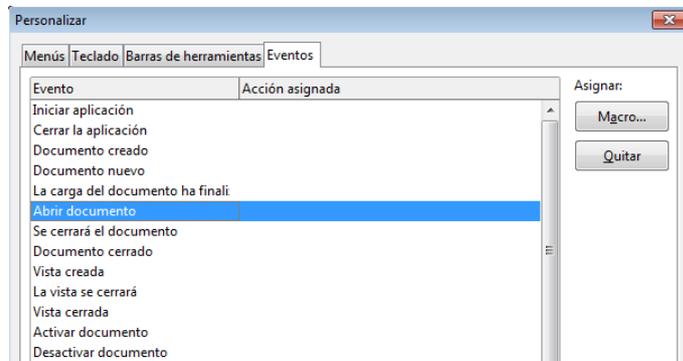
MACROS DE APERTURA

En ocasiones podemos desear que se ejecute cualquier operación al abrir una hoja de cálculo, como borrar un rango, abrir una hoja determinada, dar un valor a una celda, etc.

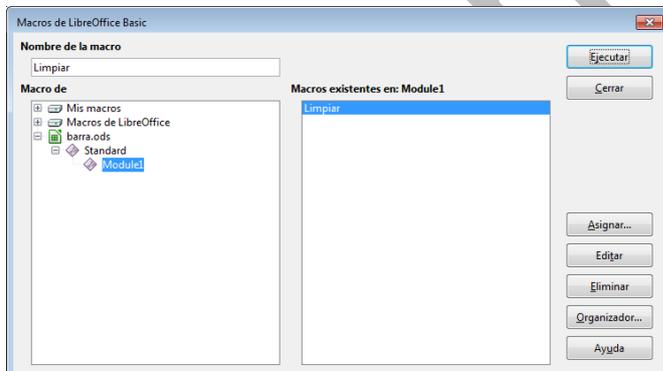
Para lograrlo debes, en primer lugar, escribir o grabar una macro con las operaciones que deseas.

Escribes o grabas una macro y le das el nombre que desees. Supongamos que fuera Sub inicio().

Una vez construida la macro, abres el menú **Herramientas - Personalizar**, eliges la pestaña Eventos y señalas con el ratón el evento de **Abrir Documento**.



Pulsas sobre el botón **Macro...**, y la buscas (por ejemplo **inicio**). Debes saber dónde está. En la imagen estaría en el documento **Sin título**, macros **Standard**, módulo **Module1**. Aceptas dos veces y ya tienes una macro de inicio.



DEFINICIÓN DE FUNCIONES

En ocasiones desearás definir funciones que la hoja de cálculo no ofrece. Por ejemplo, en Electricidad nos puede convenir definir la resistencia equivalente a otras dos situadas en paralelo, o en Geometría, la función que devuelve una hipotenusa en función de los dos catetos. Mediante un uso sencillo de las macros puedes lograrlo.

Secuencia para definir tus propias funciones

1) Abrir el Editor de Basic

Sigue el menú **Herramientas - Macros - Organizar macros - LibreOffice Basic** para abrir el editor.

Si es la primera función que defines, busca la carpeta **Standard** correspondiente al nombre de tu modelo (si lo acabas de crear, se llamará **Sin título**). No señales la otra carpeta **Standard**, que es más general. Una vez elegida la carpeta, pulsa el botón **Nuevo** para abrir un módulo contenedor. Se te ofrecerá el nombre de **module1**, **module2** u otro similar. Acepta el nombre o cámbialo según tu criterio. Al aceptar el nombre se abrirá el editor de macros. Por defecto aparecerá la macro **Main**, que puedes borrar o ignorar.

Escribe debajo el código de tu función, según se explica en el siguiente apartado.

2) Escritura del código

Terminada la secuencia anterior, comienza a escribir el código de una función.

Debes comenzar con

Public Function nombre de la función (argumento)

y terminar con

End function

y entre ambas, el código de la función

En ese código debemos usar el nombre de la función seguida del signo igual y de su definición

Es mejor verlo con un ejemplo:

```
Function cubo ( numero )  
cubo=numero*numero*numero  
End function
```

En el ejemplo, el nombre de la función es cubo, y su argumento numero (lo traduciríamos como "Cubo de un número"). Después volvemos a escribir cubo, el signo igual, y su definición.

3) Uso de la función

Una vez escrito el código, cierra el Editor de Basic y usa tu función en cualquier celda. En la imagen puedes ver una celda definida mediante la nueva función CUBO.

C5 fx =cubo(C3)			
	A	B	C
1			
2			
3		Número	4
4			
5		Su cubo	64
6			
7			

Con esto ya tienes definida la función.

Con la técnica explicada, esa función sólo estará activa en la hoja de cálculo en la que la has creado, no en otras. Al cerrar la hoja ya no podrás usarla.

Función con varios argumentos

Una función puede actuar sobre varios argumentos, por ejemplo la resistencia equivalente a la que se aludía en el primer párrafo. En ese caso, se deberán separar mediante una coma:

Function resisequiv(r1, r2)
resisequiv = r1 * r2 / (r1 + r2)
End Function

Cuando uses esta función en una celda, debes sustituir la coma por un punto y coma, por ejemplo resisequiv(4;6). Estudia el ejemplo de la imagen:

C6 fx =resisequiv(C3;C4)			
	A	B	C
1			
2			
3		Resistencia 1	4
4		Resistencia 2	6
5			
6		Resistencia equivalente	2,4
7			
8			

Variables auxiliares

En una definición puedes usar las estructuras del Basic: FOR...NEXT, SELECT CASE, etc. Aquí sólo usaremos DIM, para crear variables auxiliares. Observa este ejemplo

Function area6(lado)
Dim perimetro, apotema

perimetro = 6 * lado
apotema = lado * Sqr(3) / 2
area6 = perimetro * apotema / 2
End Function

que devuelve el área de un hexágono en función del lado. El nombre de la función, en este caso area6, debe figurar en la definición, aunque uses otras variables.