



Programación con VBA Excel



Píldoras Informáticas



CAPÍTULO 1. Introducción. Macros.....	6
1.1 Cuando utilizar una Macro	6
1.2 Grabar una macro	8
1.3 Ejecutar una Macro.....	9
1.4 Seguridad en las macros.....	11
1.5 Modificar el código de una macro.....	12
1.6 Botones de formulario para ejecutar Macros	12
CAPÍTULO 2. El Editor de Visual Basic. Lenguaje VBA.....	14
2.1 La interfaz del Editor.	14
2.1.1 Explorador de Proyectos	14
2.1.2 Propiedades.....	15
2.1.3 Código / Área de Trabajo	16
2.2 Crear un Módulo.....	¡Error! Marcador no definido.
2.3 Ayuda de VBA.....	¡Error! Marcador no definido.
Ejercicio 1. Organizando Proyectos.....	¡Error! Marcador no definido.
CAPÍTULO 3. Tipos de datos, Variables y Constantes ...	¡Error! Marcador no definido.
3.1 Tipos de datos en VBA Excel	¡Error! Marcador no definido.
3.2 Variables	¡Error! Marcador no definido.
3.3 Declaración de variables	¡Error! Marcador no definido.
3.4 Variables de tipo Array.....	¡Error! Marcador no definido.
3.5 La sentencia Option Explicit.....	¡Error! Marcador no definido.
3.6 Entrada de datos. La función Inputbox.....	¡Error! Marcador no definido.
3.7 Salida de datos. La función MsgBox	¡Error! Marcador no definido.
3.8 Funciones de conversión de tipos	¡Error! Marcador no definido.
3.9 Constantes.....	¡Error! Marcador no definido.
3.9.1 Intrínsecas o definidas por el sistema	¡Error! Marcador no definido.
3.9.2 Definidas por el Usuario	¡Error! Marcador no definido.
Ejercicio 2: Sumar dos cantidades.....	¡Error! Marcador no definido.
Ejercicio3: Petición de datos	¡Error! Marcador no definido.
CAPÍTULO 4. Construir expresiones VBA.....	¡Error! Marcador no definido.
4.1 Operadores.....	¡Error! Marcador no definido.
4.1.1 Operadores aritméticos	¡Error! Marcador no definido.
4.1.2 Operadores de concatenación	¡Error! Marcador no definido.
4.1.3 Operadores de comparación.....	¡Error! Marcador no definido.

4.1.4	Operadores lógicos.....	¡Error! Marcador no definido.
4.2	Orden de ejecución	¡Error! Marcador no definido.
4.3	Funciones matemáticas.....	¡Error! Marcador no definido.
4.4	Funciones financieras.....	¡Error! Marcador no definido.
	Ejercicio 4. Cálculos matemáticos	¡Error! Marcador no definido.

CAPÍTULO 5. Estructuras de control de flujo ¡Error! Marcador no definido.

5.1	Condicional If-Else.....	¡Error! Marcador no definido.
5.2	Condicional Select Case.....	¡Error! Marcador no definido.
5.3	Operadores lógicos	¡Error! Marcador no definido.
5.4	Estructuras repetitivas	¡Error! Marcador no definido.
5.4.1	Bucle For – Next	¡Error! Marcador no definido.
5.4.2	Estructura Do - Loop – While	¡Error! Marcador no definido.
5.4.3	Estructura Do....loop Until.....	¡Error! Marcador no definido.
5.4.4	Estructura For – Each	¡Error! Marcador no definido.
5.4.5	Sentencia While – Wend	¡Error! Marcador no definido.
	Ejercicio 5: Calcula el precio por el descuento.....	¡Error! Marcador no definido.
	Ejercicio 6: Calcula la nota media.....	¡Error! Marcador no definido.
	Ejercicio 7: Trabajo con Array.....	¡Error! Marcador no definido.

CAPÍTULO 6. Procedimientos y funciones ¡Error! Marcador no definido.

6.1	Procedimientos.....	¡Error! Marcador no definido.
6.1.1	Parámetros.....	¡Error! Marcador no definido.
6.2	Funciones	¡Error! Marcador no definido.
	Ejercicio 8: Paso de parámetros.....	¡Error! Marcador no definido.

CAPÍTULO 7. Programación Orientada a Objetos con VBA Excel..... ¡Error! Marcador no definido.

7.1	Objeto Application.....	¡Error! Marcador no definido.
7.1.1	Propiedades del objeto Application	¡Error! Marcador no definido.
7.1.2	Métodos del objeto Application.....	¡Error! Marcador no definido.
7.1.3	Recalcular libros de trabajo.....	¡Error! Marcador no definido.
7.1.4	Convertir una cadena en un objeto.....	¡Error! Marcador no definido.
7.1.5	Pausar una macro en ejecución	¡Error! Marcador no definido.
7.1.6	Métodos orientados a eventos	¡Error! Marcador no definido.
7.1.6.1	Onkey	¡Error! Marcador no definido.
7.1.6.2	OnTime	¡Error! Marcador no definido.
7.1.6.3	OnRepeat.....	¡Error! Marcador no definido.

7.2	Objeto Workbook	¡Error! Marcador no definido.
7.2.1	Especificar el objeto Workbook	¡Error! Marcador no definido.
7.2.2	Abrir un libro de trabajo.....	¡Error! Marcador no definido.
7.2.3	Crear un libro de trabajo	¡Error! Marcador no definido.
7.2.4	Cerrar un libro de trabajo.....	¡Error! Marcador no definido.
7.3	Objeto Worksheet	¡Error! Marcador no definido.
7.3.1	Propiedades de worksheet.....	¡Error! Marcador no definido.
7.3.2	Métodos de Worksheet.....	¡Error! Marcador no definido.
7.3.3	Crear una hoja de trabajo	¡Error! Marcador no definido.
7.4	Conjunto Range	¡Error! Marcador no definido.
7.5	Método Cell	¡Error! Marcador no definido.
7.5.1	Devolver una fila.....	¡Error! Marcador no definido.
7.5.2	Devolver una columna	¡Error! Marcador no definido.
7.6	Método Offset	¡Error! Marcador no definido.
7.7	Seleccionar una celda o rango	¡Error! Marcador no definido.
7.8	Definir el nombre de un rango	¡Error! Marcador no definido.
7.9	Insertar datos en el rango	¡Error! Marcador no definido.
7.10	Obtener los datos de un rango	¡Error! Marcador no definido.
7.11	Redimensionar un rango	¡Error! Marcador no definido.
	Ejercicio 9. Manejando hojas de cálculo	¡Error! Marcador no definido.
	Ejercicio 10. Evaluando celdas	¡Error! Marcador no definido.
	Ejercicio 11. Eventos	¡Error! Marcador no definido.

CAPÍTULO 8. Formularios de Usuario (UserForms) ¡Error! Marcador no definido.

8.1	Ejecutar un formulario	¡Error! Marcador no definido.
8.2	Controles	¡Error! Marcador no definido.
8.2.1	Botón de comando	¡Error! Marcador no definido.
8.2.2	Cuadro de texto.....	¡Error! Marcador no definido.
8.2.3	Etiqueta	¡Error! Marcador no definido.
8.2.4	Cuadro Combinado.....	¡Error! Marcador no definido.
8.2.5	Control checkbox.....	¡Error! Marcador no definido.
8.3	Crear un formulario	¡Error! Marcador no definido.
	Ejercicio 12. Formularios	¡Error! Marcador no definido.

CAPÍTULO 9. Técnicas útiles con VBA..... ¡Error! Marcador no definido.

9.1	Trabajar con módulos	¡Error! Marcador no definido.
------------	-----------------------------------	-------------------------------

9.1.1	Exportar un módulo	¡Error! Marcador no definido.
9.1.2	Cambiar el nombre de un módulo	¡Error! Marcador no definido.
9.1.3	Importar un módulo	¡Error! Marcador no definido.
9.1.4	Borrar un módulo	¡Error! Marcador no definido.
9.2	Acceder al sistema de archivos de VBA	¡Error! Marcador no definido.
9.2.1	DIR	¡Error! Marcador no definido.
9.2.2	Curdir	¡Error! Marcador no definido.
9.2.3	Filedatetime	¡Error! Marcador no definido.
9.2.4	Filelen	¡Error! Marcador no definido.
9.2.5	Getattr	¡Error! Marcador no definido.
9.3	Manipular el sistema de archivos con VBA.....	¡Error! Marcador no definido.
	Ejercicio 13. Accediendo al sistema de ficheros	¡Error! Marcador no definido.
CAPÍTULO 10. Manejo de errores en VBA .. ¡Error! Marcador no definido.		
10.1	Instrucción on error	¡Error! Marcador no definido.
10.2	El objeto err	¡Error! Marcador no definido.
10.2.1	Métodos del objeto Err	¡Error! Marcador no definido.
	Ejercicio 14. Manejo de errores	¡Error! Marcador no definido.
	Solucionario.....	¡Error! Marcador no definido.

CAPÍTULO 1. Introducción. Macros

Visual Basic for Applications es un lenguaje de programación común para todas las aplicaciones de Microsoft Office. En este e-book vamos a estudiar este lenguaje para el programa Excel.

La versión de Excel que se utilizará a lo largo del e-book será la 2013 si bien esto carece de importancia a la hora de programar en VBA. Si deseas utilizar una versión anterior de Excel para practicar con los conocimientos que se vayan adquiriendo, lo podrás hacer sin problemas.

Programando en VBA podemos:

- Automatizar tareas que se repiten con frecuencia en nuestro trabajo.
- Realizar múltiples acciones de búsqueda y reemplazo.
- Crear Formularios de Usuario (UserForms) para nuestros libros de trabajo.
- Modificar las opciones existentes en Excel y crear nuevas funciones (Complementos).
- Personalizar la interfaz de Excel (barra de acceso rápido y ribbon).
- Comunicar Excel con otras aplicaciones del paquete Microsoft Office.

Disponemos de dos formas para programar en VBA:

1. Utilizando la grabadora de macros.
2. Escribiendo directamente el código VBA en el Editor de Visual Basic.

Al código generado por la grabadora de macros de Excel se le denomina “macro de grabadora”.

Las macros son acciones grabadas secuencialmente que podemos ejecutar cuando ocurra cierto evento. Un evento es una acción desencadenada por el usuario o por el propio programa Excel. La funcionalidad de las macros de grabadora se basa en los elementos existentes en Excel y las circunstancias durante el proceso de grabación y por lo tanto están limitadas a las características del propio programa.

Sin embargo para crear operaciones más específicas y no estar limitados por las macros de grabadora, se deberá crear procedimientos programando manualmente en el lenguaje VBA utilizando para ello el Editor de VBA.

1.1 Cuando utilizar una Macro

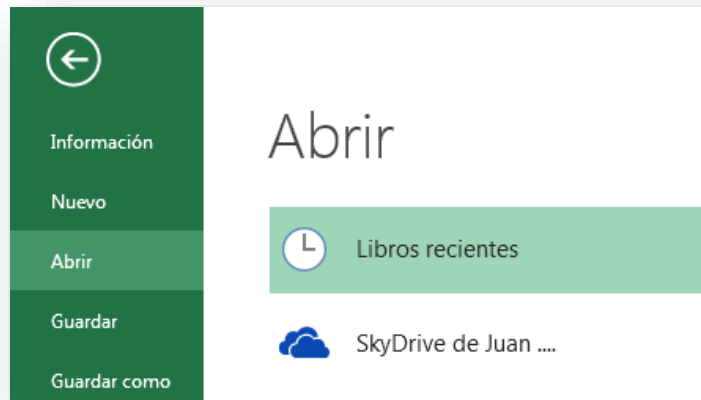
En Microsoft Excel, es posible llevar a cabo numerosas tareas mediante macros o a través de la interfaz de usuario. La elección de una macro o de programar en Microsoft Visual Basic para Aplicaciones depende generalmente de lo que se desee hacer.

Las macros son un método sencillo para controlar tareas simples.

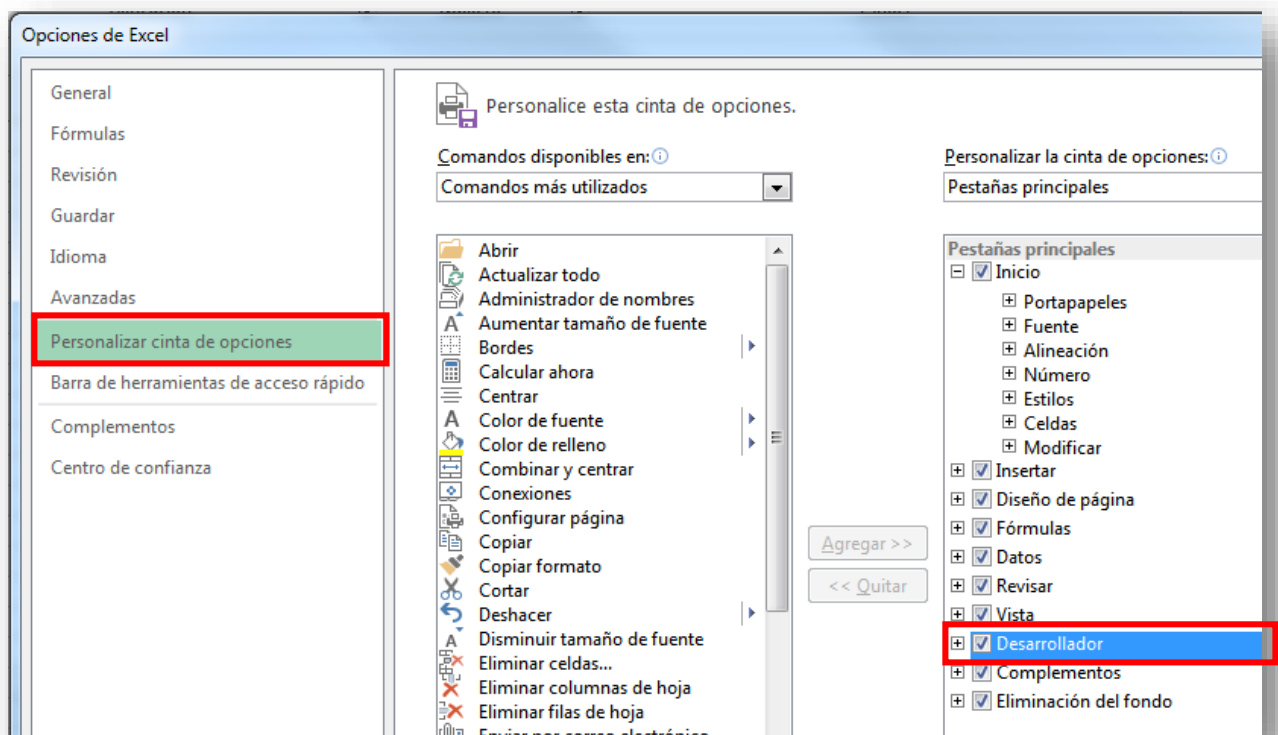
Tanto si se va a utilizar la grabadora de macros como si se van a escribir las macros de forma manual programando en VBA, es conveniente habilitar ficha **Desarrollador**.

Para activar y desactivar la ficha, seguimos los pasos siguientes:

- Hacemos clic en la ficha **Archivo** **ARCHIVO** y a continuación en **Opciones de Excel**.



Se muestra la ventana **Opciones de Excel**:



- Una vez activadas las opciones resaltadas en las imágenes, hacer clic en el botón **Aceptar**.

1.2 Grabar una macro

La forma más fácil e intuitiva de crear macros es crearlas mediante el grabador de macros. El grabador de macros mientras realizamos las tareas deseadas simultáneamente va traduciendo las mismas a instrucciones VBA que podemos modificar.

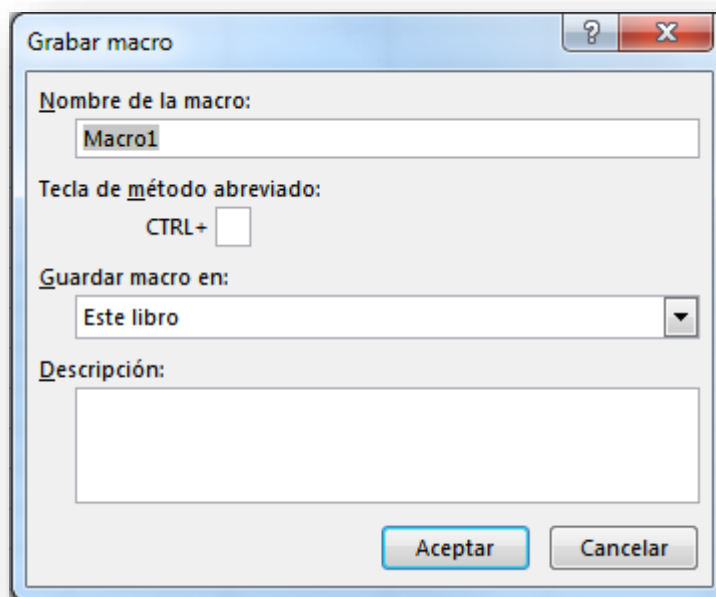
Antes de grabar una macro, podemos definir el tipo de referencias que usaremos mediante el botón **Usar referencias relativas**. En el modo Referencias Absolutas los rangos de celdas referenciados en la grabación de la macro serán fijos, este es el valor por defecto, sin activar el botón. En caso de seleccionar **Usar referencias relativas**, los rangos de celdas serán expresados en relación a la posición de la celda activa.



Para grabar una macro desde la ficha **Programador/Desarrollador**, hacemos clic en el botón **Grabar macro**.



Se muestra la ventana **Grabar Macro**:



Podemos asignar un nombre a la macro, en el campo **Nombre de la macro**. No está permitido insertar ni espacios en blanco ni caracteres especiales tales como símbolos de interrogación, exclamación o similares en el nombre de la macro.

Se puede asignar una combinación de teclas que ejecute la macro en el campo **Método abreviado**, combinando la tecla **Control** con alguna otra tecla, teniendo la precaución de no elegir una combinación que utilicemos para otro propósito, como por ejemplo CTRL+C (copiar) o CTRL+V (pegar), ya que de esta forma la acción por defecto quedaría invalidada por la macro.

En **Guardar macro en**, podemos seleccionar guardar la macro en el libro activo, en el libro de macros personal o en un libro nuevo. Si seleccionamos el libro de Macros personal, la macro se graba en el libro “personal.xlsm” y será accesible desde todos los libros de Excel.

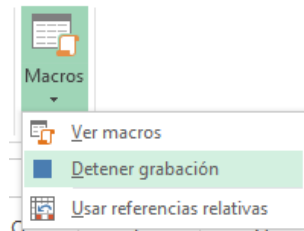
En **Descripción**, podemos describir cuál es el cometido de la macro o cualquier otro dato que creamos conveniente.

Para comenzar la grabación de la macro haremos clic en el botón **Aceptar** y a continuación, se muestra en la barra de estado un botón que indica que se está grabando la misma.



Este botón nos permite detener la grabación de la macro. Desde este momento se deben realizar las acciones que queramos grabar. Una vez terminadas las acciones, podemos también detener la grabación haciendo clic en el botón **Detener grabación** de la ficha **Programador / Desarrollador** o también desde la ficha **Vista**.

■ Detener grabación



1.3 Ejecutar una Macro

Una vez creada una macro, la podemos ejecutar las veces que queramos. Para ejecutar la macro hay varias alternativas:

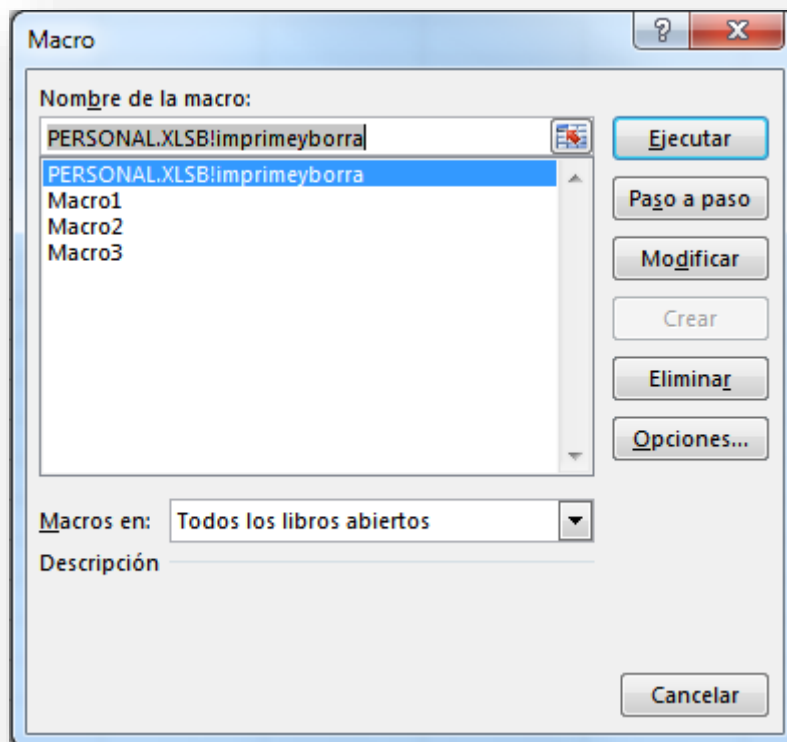
- Utilizar la combinación de teclas escogida en la ventana de **Grabar Macro**.
- Desde la ventana de **Macros**.
- Haciendo clic en un botón de formulario.
- Haciendo clic en un botón de macro de la **barra acceso rápido**.
- Haciendo clic en un botón de macro de la **cinta de opciones** (solo para las versiones 2010 y 2013).

Para ejecutar la macro desde la ventana de Macros:

- Hacer clic en el botón Ver Macros



- Se muestra el cuadro de diálogo Macro que presenta una lista con las macros existentes.



- Seleccionamos la macro y hacemos clic en el botón **Ejecutar**.
- **Cancelar** - Cierra el cuadro de diálogo sin realizar ninguna acción.
- **Paso a paso** - Ejecuta la macro instrucción a instrucción desde el editor de Visual Basic.

```

PERSONAL.XLSB - Módulo2 (Código)
(General) | imprimeyborra
Sub imprimeyborra()
'
' imprimeyborra Macro
'
' Acceso directo: CTRL+y
'
Application.DisplayAlerts = False
Range("A1:D3").Select
Charts.Add
ActiveChart.ChartType = xlColumnClustered
ActiveChart.SetSourceData Source:=Range("Hoja1!$A$1:$D$3")
Sheets("Gráfico1").Select
ActiveChart.ChartArea.Select
Sheets("Gráfico1").Select
ActiveWindow.SelectedSheets.Delete
Range("I18").Select
End Sub

```

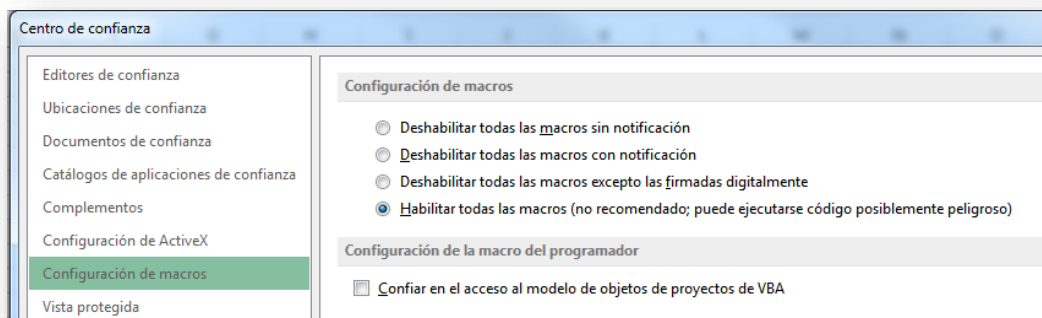
- **Modificar** - Abre el editor Visual Basic para modificar el código de la macro.

- **Eliminar** - Borra la macro.
- **Opciones** - Abre otro cuadro de diálogo donde podemos modificar la tecla de método abreviado y la descripción de la macro.

1.4 Seguridad en las macros

La configuración de seguridad nos permite controlar los libros que contienen macros. Para configurar la seguridad hacemos clic en el botón **Seguridad de macros**.

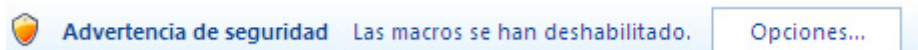
⚠ Seguridad de macros



Se muestra el cuadro de diálogo **Centro de confianza**:

Las opciones de configuración de macros que podemos utilizar son las siguientes:

- **Deshabilitar todas las macros sin notificación**, todas las macros y todas las advertencias de seguridad serán deshabilitadas sin excepción.
- **Deshabilitar todas las macros con notificación**, opción por defecto, las macros serán deshabilitadas pero se muestra una advertencia de seguridad. Para los libros con macros.



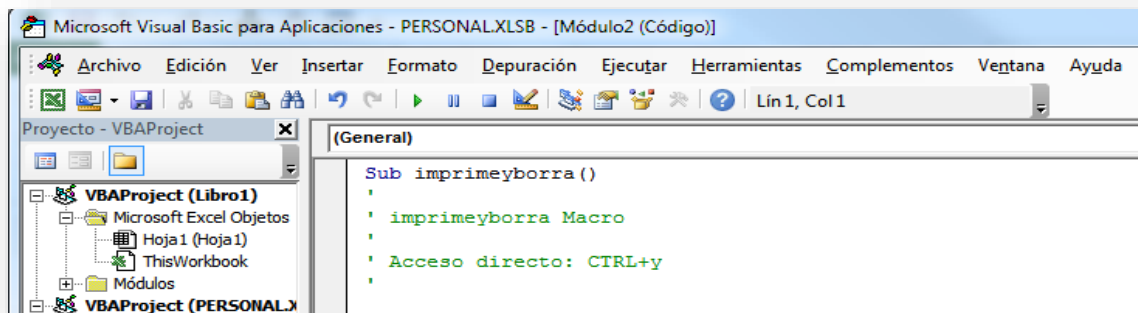
- **Deshabilitar todas las macros excepto las firmadas digitalmente**, esta configuración es igual que Deshabilitar todas las macros con notificación excepto en que si la macro está firmada digitalmente por un editor de confianza, puede ejecutarse si ya se ha confiado en el editor previamente. De lo contrario, se le notifica con una advertencia de seguridad. De este modo, puede decidir habilitar esas macros firmadas o confiar en el editor. Todas las macros sin firmar se deshabilitan sin notificación.
- **Habilitar todas las macros**, esta opción permite que se ejecuten todas las macros. Esta configuración hace que el equipo sea vulnerable a código posiblemente malintencionado y no se recomienda. Hay que tener especial cuidado con esta opción activada de conocer la procedencia de los archivos con macros asociadas.
- **Confiar en el acceso al modelo de objetos de proyectos de VBA**, esta configuración es para programadores de VBA. Se permite ejecutar macros que accedan a objetos pertenecientes a la biblioteca VBA. Esta opción se entenderá mejor cuando hayamos llegado al capítulo de POO (Programación Orientada a Objetos)

1.5 Modificar el código de una macro

Como mencionamos anteriormente podemos acceder al código de un macro haciendo clic en el botón **Macros** de la ficha **Programador**.



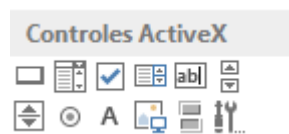
Elegimos la macros que queremos modificar y hacemos clic en el botón **Modificar**. Se abre el editor de VBA donde podremos re-programar el código de una macro, corrigiendo borrando o agregando nuestro propio código. Más adelante, estudiaremos con más detalle el mismo.



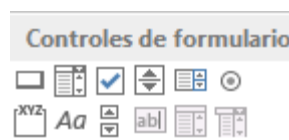
1.6 Botones de formulario para ejecutar Macros

Excel dispone de un conjunto de elementos (botones) de control o de formulario a los cuales podemos asignar la ejecución de una macro. Estos botones de control son de dos tipos:

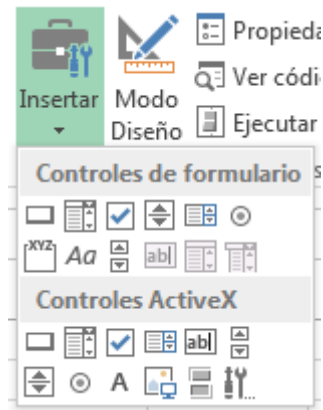
- **Controles ActiveX**, que conforman la barra de herramientas del **Cuadro de Controles**.



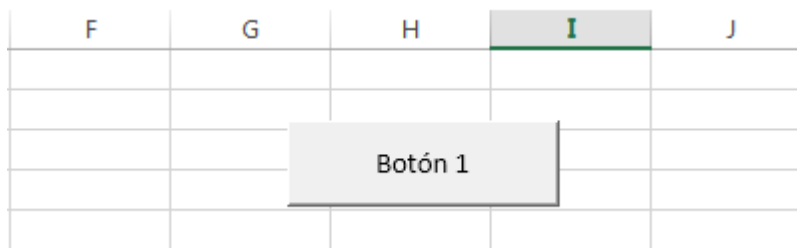
- **Controles de formulario**.



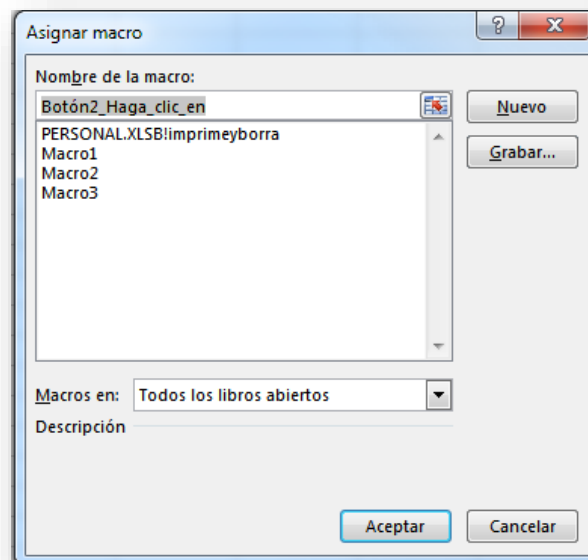
Para acceder a estos controles en la ficha **Desarrollador / Programador**, hacemos clic en el botón **Insertar**.



Para asignar una macro a un botón, hacemos clic en el botón “Botón” del grupo de controles de formulario, arrastramos en el lugar de la hoja donde queremos insertar el elemento.



Tras soltar el botón se abre un cuadro de dialogo desde el cual seleccionamos la macro que queremos asignar. Seleccionamos la macro y hacemos clic en el botón **Aceptar**.



Para finalizar bastará hacer clic sobre el botón para ejecutar la macro.

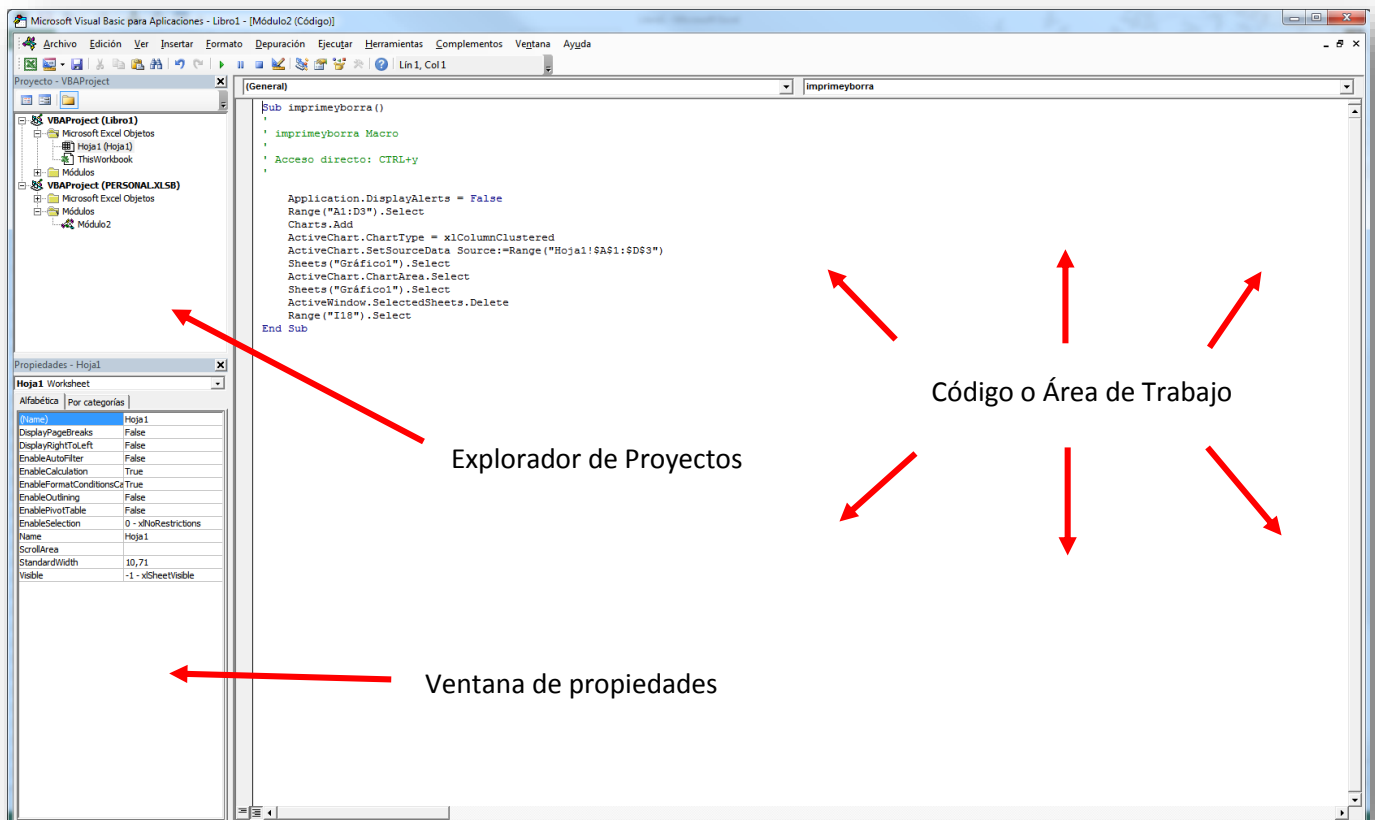
CAPÍTULO 2. El Editor de Visual Basic. Lenguaje VBA

El Editor de Visual Basic es la herramienta que nos permite ampliar la eficacia y versatilidad de las macros más allá la propia grabación. ¿Cómo?: Editando y modificando las macros previamente grabadas o escribiendo nuestro propio código.

2.1 La interfaz del Editor.

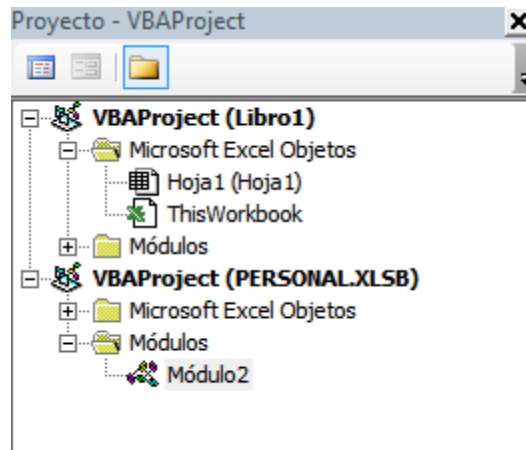
El Editor de Visual Basic muestra diferente información en distintas ventanas. Las principales ventanas son:

- Explorador del proyecto.
- Propiedades.
- Código o Área de Trabajo.



2.1.1 Explorador de Proyectos

La ventana Explorador de Proyectos nos permite seleccionar la macro en la que se desea trabajar. El Explorador de Proyectos muestra todos los proyectos (grupos de códigos) y las macros que contienen en una vista en forma de árbol. La raíz u objetos de base de la estructura de árbol son los libros actuales (Proyectos) en los que se está trabajando, y los demás libros y complementos que estén abiertos.

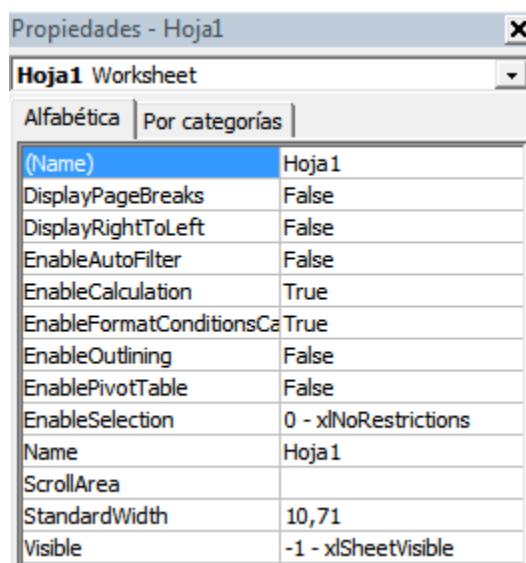


Cada uno de los elementos principales que se utilizan en los proyectos VBA se guarda en una carpeta independiente en cada proyecto. Estos elementos son:

- **Objetos.** Las hojas del libro abierto y el propio libro (ThisWorkBook).
- **Módulos,** que contienen el código de macro asociado al libro.
- **Módulos de clase,** que son objetos definidos y creados por el usuario / programador.
- **Formularios de usuario (UserForm),** que ofrecen un espacio en el que se pueden colocar controles gráficos, como botones, imágenes y áreas de texto.

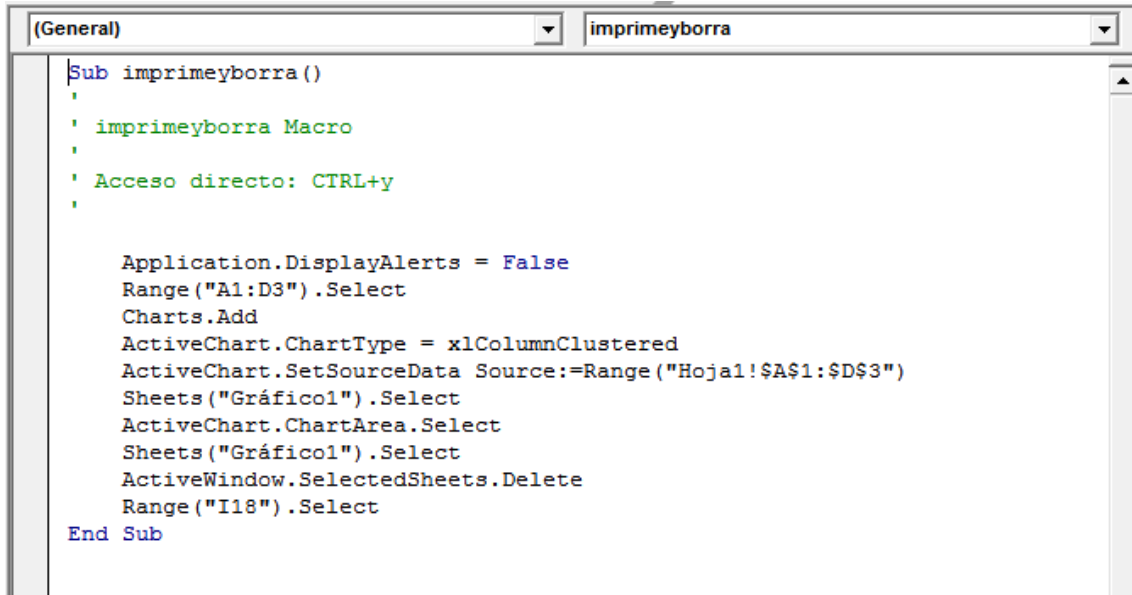
2.1.2 Propiedades

Este panel se utiliza para examinar y modificar las distintas propiedades asociadas al objeto seleccionado. La única propiedad que suele estar disponible para los módulos es el nombre. Las hojas de cálculo tienen propiedades adicionales que se pueden modificar, como ScrollArea y Visible. Se utiliza mucho cuando se trabaja con Formularios de Usuario (UserForm).



2.1.3 Código / Área de Trabajo

La ventana **Código** en el área central del Editor de VBA incluye en su lado superior dos cuadros desplegables. El cuadro desplegable de la izquierda (cuadro Objeto) se utiliza para seleccionar el objeto en el que se quiere trabajar. Cuando se trabaja sólo con código, el cuadro muestra el objeto General predeterminado. El segundo cuadro desplegable (cuadro Procedimiento) se utiliza para navegar por los procedimientos del módulo actual. Según se agreguen o eliminen macros en el módulo, se agregarán y eliminarán en el cuadro Procedimiento. Cuando se trabaja con POO (Programación Orientada a Objetos) este desplegable muestra los eventos del objeto seleccionado.



The screenshot shows the VBA Code Editor window. At the top, there are two dropdown menus: the left one is set to '(General)' and the right one is set to 'imprimeyborra'. The main area contains the following VBA code:

```
Sub imprimeyborra()
'
' imprimeyborra Macro
'
' Acceso directo: CTRL+y
'
Application.DisplayAlerts = False
Range("A1:D3").Select
Charts.Add
ActiveChart.ChartType = xlColumnClustered
ActiveChart.SetSourceData Source:=Range("Hoja1!$A$1:$D$3")
Sheets("Gráfico1").Select
ActiveChart.ChartArea.Select
Sheets("Gráfico1").Select
ActiveWindow.SelectedSheets.Delete
Range("I18").Select
End Sub
```

Utilizando las teclas de las flechas, podemos desplazarnos por las listas del código y buscar la macro. Conforme se vaya desplazando por el código, el cuadro Procedimiento se irá actualizando para mostrar la macro actual.