

Material Imprimible

Curso de Excel Macros

## Módulo 5: Programando sobre formularios

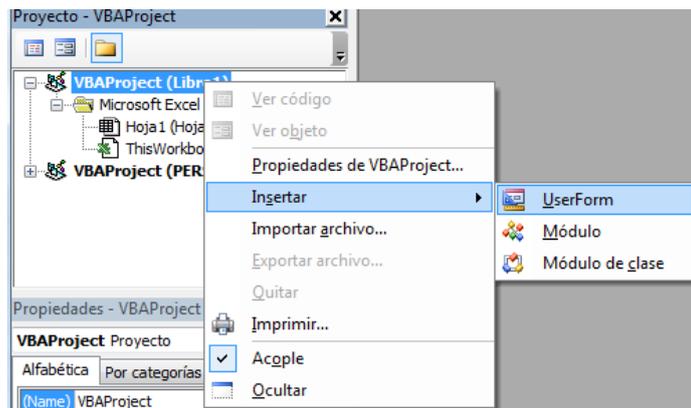
### Formularios

Un formulario es un cuadro de diálogo que nos permite interactuar con el usuario. Es un cuadro que posee distintos elementos mediante los cuales el usuario recibe o brinda información a la macro.

Los elementos que componen un formulario se llaman controles.

Los formularios (UserForm) se crean dentro de los proyectos que visualizamos en el Explorador, en una carpeta específica de Formularios. Si queremos acceder a ellos debemos seleccionar previamente el proyecto en el que se encuentran almacenados.

Para crear un formulario en el Editor de Visual Basic tendremos que hacer click derecho sobre el título del proyecto y dentro del menú que se despliega seleccionar la opción Insertar y posteriormente UserForm.



También podremos crearlo desde el menú Insertar de la barra de menús (primero tendremos que seleccionar el proyecto al cual le agregaremos este módulo). Insertado el primer formulario se creará la carpeta Formularios dentro de nuestro proyecto en donde podremos visualizar todos ellos.

### Vistas

Los formularios se pueden abrir en vista OBJETO y en vista CÓDIGO. Haciendo click sobre éste podremos entrar a cualquiera de las opciones.

Si se abren en vista objeto, se podrá ver la apariencia y presentación del formulario.

Si se abren en vista código, se podrán ver los comandos asociados a los eventos generados por el formulario (por ejemplo, al hacer click en un botón o al elegir una opción).

### Vista objeto

En esta vista se podrá ver la apariencia y presentación del formulario. Luego le daremos funcionalidad en la vista código.

Desde aquí, trabajaremos:

- Insertar elementos (controles) al formulario para interactuar con el usuario.
- Apariencia estética del formulario.
- Orden de tabulación.

### Insertar controles. Apariencia estética

Los controles son elementos que se pueden agregar a un formulario para interactuar con el usuario. Es decir, que el usuario pueda leer o escribir una información.

Para visualizar y poder agregar estos controles, tendremos que tener abierto el cuadro de herramientas. Si no estuviera en pantalla, se lo podrá activar haciendo click en el **Menú VER >> Cuadro de herramientas**, en el editor de Visual Basic.

Luego, en ese Cuadro de herramientas, seleccionaremos el control que deseamos insertar y lo arrastraremos hacia el formulario. A partir de allí, se le podrán editar sus propiedades en la Ventana de propiedades. Si no estuviera en pantalla, se la podrá activar haciendo click en el **Menú VER >> Ventana de propiedades**. También se lo podrá mover de lugar y cambiar de tamaño fácilmente con el mouse.

Desde la ventana de propiedades se manejará la apariencia estética del formulario y de cada uno de sus elementos: nombre, título, color de letra, color de fondo, etcétera. Si se selecciona el formulario, se cambiarán las propiedades del formulario; si se selecciona un control, se le cambiarán las de este control.

### Listado de controles:

#### Seleccionador de objetos



Permite seleccionar los controles que hemos insertado en el Formulario.

### **Etiqueta (LABEL)**

 Sirve para introducir un título, una descripción o, en definitiva, cualquier tipo de texto para ubicarlo en el Formulario

### **Cuadro de texto (TEXTBOX)**

 Sirve para que el usuario pueda introducir un dato, sea numérico o sea texto.

### **Cuadro combinado (COMBOX)**

 Sirve para que el usuario elija una opción de la lista. El control mostrará la opción seleccionada y desplegará el resto de las opciones cuando el usuario haga click en la flecha. Sólo se puede seleccionar una opción.

### **Cuadro de lista (LISTBOX)**

 Sirve para que el usuario elija una o varias opciones de la lista. El control mostrará varias opciones, según su tamaño, para que el usuario seleccione las que desee.

### **Casilla (CHECKBOX)**

 Sirve para que el usuario seleccione una o varias opciones a través del tildado de la casilla.

### **Botón de opción (OPTIONBUTTON / RADIOBUTTON)**

 Sirve para que el usuario seleccione una sola opción de entre varias posibilidades.

### **Botón de alternar (TOGGLEBUTTON)**

 Sirve para activar o desactivar alguna funcionalidad del Formulario. El control admite dos estados: “encendido” y “apagado”.

### **Marco (FRAME)**

 Sirve para agrupar varios controles de un formulario. A través de este, por ejemplo, podremos encerrar todas las opciones de un botón de opción para que el usuario sólo pueda elegir una.

### **Botón de comando (COMMANDBUTTON)**

 Sirve para ejecutar una o varias acciones al hacer click sobre él.

### **Barra de tabulaciones (TABSTRIP)**



Sirve para crear distintas secciones dentro de un Formulario.

### **Página Multiple (MULTIPAGE)**



Sirve para crear varias páginas dentro de un Formulario. Cada página será visualizada a través de una etiqueta que contenga su nombre y, adentro de ella, se podrán colocar distintos controles que sólo serán visibles cuando se la seleccione.

### **Barra de desplazamiento (SCROLLBAR)**



Sirve para navegar a través del formulario cuando éste tuviera distintos controles no visibles en un mismo momento por su tamaño.

### **Botón de número (SPINBUTTON)**



Sirve para aumentar o disminuir valores.

### **Imagen (IMAGE)**



Sirve para introducir una imagen en el Formulario.

### **Referencia a celda (REFEDIT)**



Permite hacer referencia a una celda.

## **Propiedades básicas del formulario y de los controles (de la Ventana de propiedades):**

**Name:** Es el nombre del formulario o del control seleccionado (nos será útil para usar en la programación).

**Caption:** Es el título que se muestra en pantalla del formulario o del control seleccionado.

**Value:** Es el valor que se escribe como dato en el control seleccionado.

**ForeColor:** Es el color de letra del formulario o del control seleccionado.

**BackColor:** Es el color de relleno del formulario o del control seleccionado.

**BorderColor:** Es el color de borde del formulario o del control seleccionado.

**Font:** Es la configuración de la fuente del formulario o del control seleccionado.

**Height:** Es la altura del formulario o del control seleccionado.

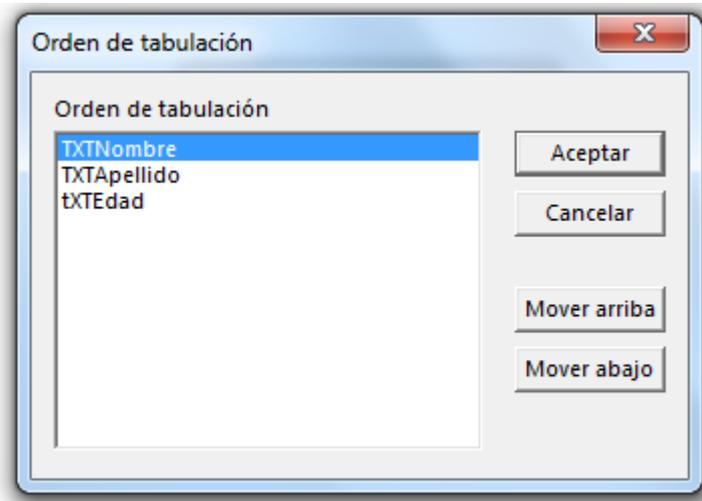
**Width:** Es el ancho del formulario o del control seleccionado.

**Enabled:** Es la propiedad que define si el formulario o el control está habilitado para que el usuario ingrese o modifique un dato. Si la propiedad es *True*, está habilitado. Si es *False*, no está habilitado.

### Orden de tabulación

El orden de tabulación es el orden en que el usuario que esté utilizando un formulario va a pasar de un control a otro rápidamente si toca la tecla del Tabulador. Con esta tecla, no necesitaríamos el mouse para seleccionar un control u otro del mismo formulario.

Para acceder y ver el orden de tabulación, haremos click en el **Menú VER >> Orden de tabulación**. En la ventana que aparezca, podremos indicarle al Editor en qué orden se pasará de un control a otro con la tecla Tabulador. Para eso, contamos



con los botones de “Mover arriba” y “Mover abajo” el control elegido.

Para trabajar cómodamente con el orden de tabulación, será necesario que cada control tenga un nombre claro que me permita identificarlo rápidamente.

Normalmente a los controles se les pone como nombre: el tipo de control + el dato que se escribirá. Por ejemplo, “TXTNombre” representa con control de Cuadro de Texto (TEXTBOX o TXT en inglés) en donde se escribirá el NOMBRE del elemento que estemos trabajando desde nuestro formulario.

Nota: A algunos elementos, como las etiquetas, al no ser editables por el usuario no se los podrá seleccionar con la tecla del Tabulador.

## Vista código

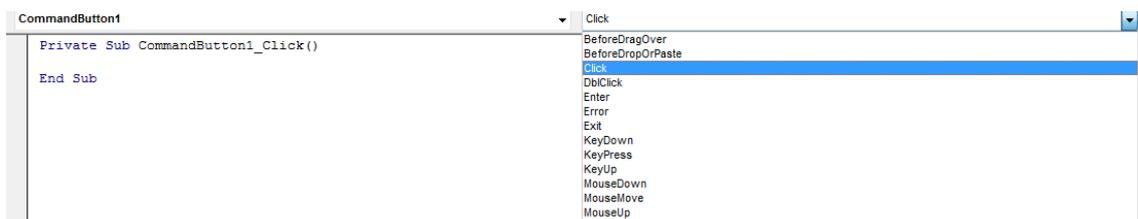
En esta vista se podrán ver los comandos asociados a los eventos generados por el formulario. Aquí le estaremos dando funcionalidad al formulario diseñado en vista objeto.

En este espacio escribiremos los comandos que se producirán cuando ocurra algún evento respecto de algún control del formulario, por ejemplo, al hacer click en un botón.

El conjunto de comandos se escribirá como una subrutina con un nombre específico. Por ejemplo: **Private Sub CommandButton1\_Click()** es el código inicial de la subrutina que se ejecutará cuando “se haga click en el botón CommandButton1”.

No hace falta saber cómo se escribe cada evento. Con el formulario en vista código, seleccionaremos el evento que necesitamos en la parte superior de la ventana.

Del lado izquierdo, elegir el control solicitado y del lado derecho, el evento deseado.



Dentro de esa subrutina escribiremos los comandos.

Sin embargo, también hay otra manera de escribir rápidamente el nombre de la subrutina. Al hacer **dobles click** sobre el control respectivo se abrirá el Formulario en **Vista Código** para poder **introducir las instrucciones** que se ejecutarán **cuando suceda un evento relacionado** a éste.

## Relación comandos del Módulo y comandos del Formulario

En el módulo se escribirán las macros que se ejecutarán cuando el usuario se lo pida al programa de Excel.

En el Formulario (en vista código) se escribirán las macros que se ejecutarán cuando ocurra un evento dentro del formulario.

Desde el módulo, para realizar alguna acción con el formulario tendremos que indicar su nombre seguido de la acción (método) deseado.

Código para mostrar el Formulario desde el Módulo:

**NOMBREDELFORMULARIO.SHOW**

CÓDIGO PARA ESCONDER EL FORMULARIO DESDE EL MÓDULO:

**NOMBREDELFORMULARIO.HIDE**

CÓDIGO PARA DESCARGAR EL FORMULARIO DESDE EL MÓDULO:

**UNLOAD NOMBREDELFORMULARIO**

Pero, desde el formulario, para realizar alguna acción con éste podremos indicar la palabra "Me" (que en inglés hace referencia a sí mismo) seguido de la acción (método) deseado.

Código para esconder el Formulario desde el propio formulario (en vista código):

**ME.HIDE**

CÓDIGO PARA DESCARGAR EL FORMULARIO DESDE EL PROPIO FORMULARIO (EN VISTA CÓDIGO):

**UNLOAD ME**

Desde el módulo, para trabajar con un control del formulario, tendremos que recurrir a la jerarquía.

**NOMBREDELFORMULARIO.NOMBRECONTROL.VALUE**

Pero desde el mismo formulario (en vista código), para trabajar con ese control, podremos simplemente mencionar su nombre.

**NOMBRECONTROL.VALUE**