

# 1. Formularios.

## 1.1 Introducción.

Los formularios son los objetos que catalizan las aplicaciones Windows, son un contenedor donde colocamos el resto de objetos que componen nuestro programa.

Los objetos básicamente son los mismos, hay de nuevos, alguno ha desaparecido, y otros han cambiado de nombre o se han desdoblado.

Los formularios siguen siendo objetos que dependen de System.Windows.Forms.

La información que figura en la mayoría de este tema, ha sido extraída de la documentación de Visual con el fin de dar una vista rápida de los cambios entre la versión 6 y el 2005, pero puede haber sufrido cambios.

## 1.2 Objetos.

Hay muchos cambios en cuanto a los objetos, controles, actuales en Visual Basic, algunos han desaparecido, pero han sido sustituidos por otros con mejores prestaciones. También han aparecido otros nuevos, que así a simple vista están bien.

Los controles pertenecen a System.Windows.Forms, donde encontraremos todos los controles estándar dentro de los formularios de Visual.

### 1.2.1 Han desaparecido

Image.  
Shape.  
Line.

### 1.2.2 De texto.

Label.  
TextBox.  
LinkLabel.  
RichTextBox.

### 1.2.3 De selección.

RadioButton, OptionButton antes.  
CheckBox.

### 1.2.4 Listas.

ListBox.  
ComboBox.  
CheckedListBox, antes una opción de ListBox.

### 1.2.5 Acción.

Button, antes CommandButton.

### 1.2.6 Agrupación.

GroupBox, antes era Frame.  
Panel, igual que el GroupBox pero sin texto.  
PictureBox.  
TabControl.

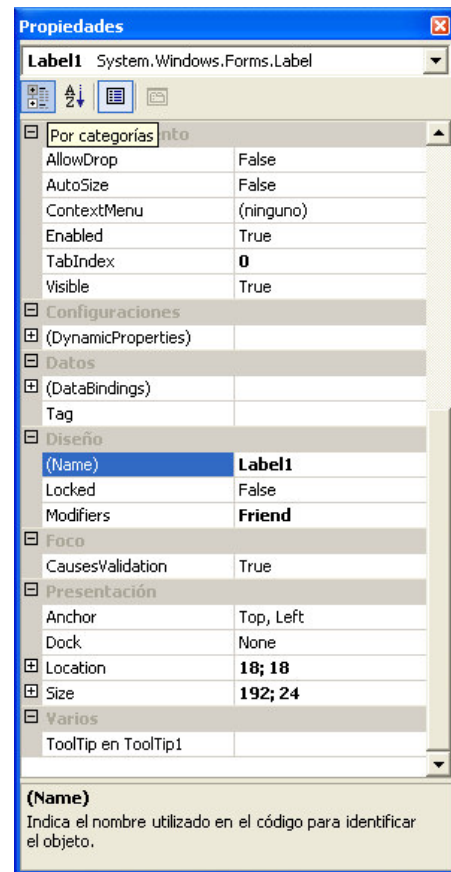
### 1.2.7 Navegación.

ListView  
TreeView  
ImageList

### 1.2.8 Cuadros de diálogo.

Antes estaban agrupados en el objeto CommonDialog, ahora se han independizado, y algunos son nuevos.

OpenFileDialog	SaveFileDialog.	FolderBrowserDialog.
FontDialog	ColorDialog	PrintDialog.
PrintPreviewDialog	PrintDocument.	PageSetupDialog.



### 1.2.9 Varios.

Timer  
DateTimePicker  
MonthCalendar  
MainMenu.  
Splitter.  
ErrorProvider.

### 1.2.10 Desplazamiento.

HscrollBar  
VscrollBar  
TrackBar  
ProgressBar

## 1.3 Propiedades.

Evidentemente, aparecen muchas nuevas, no porque lo sean quizás, sino porque se han hecho accesibles por el cambio en la filosofía del lenguaje, y antes no lo eran, y otras han cambiado de nombre.

La ventana puede distribuirse por categorías o alfabéticamente.

Hay una propiedad nueva que mejora la apariencia de un objeto cuando el ratón pasa por encima, que es FlatStyle.

### 1.3.1 Desaparecen.

DataChanged  
DataField  
DataFormat  
DataMember  
DataIcon  
DataSource  
DragIcon  
DragMode  
LinkItem  
LinkMode  
LinkTimeOut  
LinkTopic

### 1.3.2 Han cambiado.

Text, en muchos casos sustituye a la propiedad caption.

Top, ha cambiado por Y, y Top es para el uso en Anchor.

Hwnd ha cambiado por Handle.

Left, ha cambiado por X, y Left es para el uso en Anchor.

Location, X,Y, agrupa a las dos anteriores con el formato, Location X;Y

Locked por ReadOnly.

### 1.3.3 Nuevas

AcceptsReturn	Obtiene o establece un valor que indica si, al presionar <Intro> en un control TextBox multilínea, se crea una nueva línea de texto en el control o se activa el botón predeterminado del formulario.
AcceptsTabs	Obtiene o establece un valor que indica si al presionar la tecla <TAB> en un control de cuadro de texto multilínea se escribe un carácter TAB en el control en lugar de moverse el foco al siguiente control en el orden de tabulación, TabIndex.
AllowDrop	Disponible o no la recepción de eventos de arrastrar y pegar.
Anchor	Delimitan a que lado se ha de acoplar el control, izquierdo o derecho.
Appearance	Si se visualiza normal o como botón el checkBox, y RadioButton.
AutoCheck	La casilla cambia de estado al seleccionarla.
BackColorImage	La imagen de fondo de un control.
CausesValidation	Indica si el control genera o no eventos de validación.
CharacterCasing	Obtiene o establece si el control TextBox modifica la condición de mayúscula o minúscula de los caracteres a medida que se escriben.
CheckAlign	Permite alinear el checkbos al centro, izq. y derecha.
Checked	Si esta o no activado el control checkBox
CheckState	Si esta unchecked o un nuevo valor indeterminado.

ContextMenu	Menú contextual que se muestra cuando el usuario hace clic con el botón derecho del ratón.
ControlBox	Determina si el formulario tiene un cuadro de menú, control o sistema.
Cursor	El icono del ratón al pasar por encima del control.
Dock	Ubicación de acoplamiento del control, indica hacia que lado ha de estar acoplado el control.
FlatStyle	El estilo que toma el control cuando el ratón pasa por encima.
HideSelection	Obtiene o establece un valor que indica si el texto seleccionado en el control de cuadro de texto continúa resaltado cuando el control pierde foco.
Image	Permite asignar la imagen.
ImageAlign	Indica su alineación en el control.
ImeMode	Obtiene o establece el modo de Editor de métodos de entrada (IME) del control.
Lines	Obtiene o establece las líneas de texto de un control de cuadro de texto.
Modifiers	Indica el nivel de visibilidad del objeto.
ReadOnly	Indica si se puede o no cambiar el texto en el control.
RightToLeft	Indica si el control debe dibujar de izquierda a derecha o al revés, en función del idioma.
Size	Agrupar a Width y Height, en éste orden, Size Width;Height
TextAlign	El mismo significado que Aligment pero con posibilidades en vertical además de horizontal.
UseMnemonic	Significa que el carácter que vaya precedido del "&", será usado como acceso rápido con la combinación de la tecla <Alt> + ...

Además hay unas propiedades nuevas para definir el funcionamiento y contenido de los valores de accesibilidad para las personas con discapacidad. AccessibleDescription, AccessibleRole, AccesibleName.

### 1.3.4 No cambian.

Width	Height	Name	BackColor, ForeColor
BorderStyle	Enabled	Font	Locked
MaxLength	Multiline	PasswordChar	ScrollBars
TabIndex	Tag	Visible	WordWrap

## 1.4 Eventos.

### 1.4.1 Introducción.

En los eventos los cambios que ha habido es un dislate, creemos que es esa la palabra, pues las propiedades más o menos son muy parecidas, pero los eventos ....

Un evento es una acción a la que se puede responder o que se puede "controlar" mediante código.

Los eventos se pueden generar mediante una acción del usuario, por ejemplo, haciendo clic con el *mouse* (ratón) o presionando una tecla, mediante código del programa o mediante el sistema.

Las aplicaciones controladas por eventos ejecutan código en respuesta a los eventos. Cada formulario y cada control expone un conjunto de eventos predefinido que se puede utilizar al programar. Si se produce uno de estos eventos y hay código en el controlador de eventos asociado, se invocará este código.

Los tipos de eventos que produce un objeto varían, pero muchos tipos son comunes a la mayoría de los controles. Por ejemplo, la mayoría de los objetos controlan un evento **Click**: si un usuario hace clic en un formulario, se ejecutará el código del controlador de eventos **Click** del formulario.

Muchos eventos se producen junto con otros eventos. Por ejemplo, mientras se produce el evento **DoubleClick**, se producen también los eventos **MouseDown**, **MouseUp** y **Click**.

### 1.4.2 Los delegados y su función.

Los delegados son clases que se suelen utilizar dentro de .NET Framework para construir mecanismos de control de eventos. Equivalen de forma aproximada a los punteros a función, que se suelen utilizar en C++ y en otros lenguajes orientados a objetos. No obstante, a diferencia de los punteros a función, los delegados están orientados a objetos, proporcionan seguridad de tipos y son seguros. Además, mientras que un puntero de función contiene sólo una referencia a una función concreta, un delegado consta de una referencia a un objeto y hace referencia a uno o más métodos dentro del objeto.

Este modelo de eventos utiliza *delegados* para enlazar eventos a los métodos utilizados para controlarlos. El delegado permite que se registren otras clases para la notificación de eventos mediante la especificación de un método controlador. Cuando se produce el evento, el delegado llama al método enlazado.

Los delegados pueden enlazarse a un solo método o a múltiples métodos, lo que se conoce como multidifusión. Cuando se crea un delegado para un evento, el programador o el Diseñador de Windows Forms crea un evento de multidifusión. Una rara excepción puede ser un evento cuyo resultado sea un procedimiento específico (tal como la presentación de un cuadro de diálogo) que no se repita lógicamente varias veces por cada evento.

Un delegado de multidifusión mantiene una lista de llamadas de los métodos a los que está enlazado. El delegado de multidifusión admite un método Combine para agregar un método a la lista de llamadas y un método Remove para quitarlo.

Cuando la aplicación registra un evento, el control produce un evento mediante una llamada al delegado del evento. El delegado, a su vez, llama al método enlazado. En el caso más habitual (un delegado de multidifusión), el delegado llama por orden a cada uno de los métodos enlazados de la lista de llamadas, lo que proporciona una notificación de uno a muchos. Esta estrategia significa que el control no necesita mantener una lista de objetos de destino para la notificación del evento: el delegado controla todo el proceso de registro y notificación.

Los delegados permiten también enlazar múltiples eventos al mismo método, lo que permite la notificación de muchos a uno. Por ejemplo, tanto un evento clic de botón como un evento clic de comando de menú pueden invocar al mismo delegado que, a continuación, llama a un solo método para controlar de la misma manera estos eventos separados.

El mecanismo de enlace que se utiliza con los delegados es dinámico: un delegado puede enlazarse en tiempo de ejecución a cualquier método cuya firma coincida con la del controlador de eventos. Esta función permite configurar o cambiar el método de enlace dependiendo de una condición y asociar dinámicamente un controlador de eventos a un control.

### 1.4.3 Desaparecen

DragDrop	DragOver	LinkClose
LinkError	LinkNotify	LinkOpen
OleCompleteDrag	OleDragDrop	OleDragOver
OleGiveFeedBack	OleSetData	OleStartDrag

### 1.4.4 Genéricos.

#### Por cambios de valor de propiedades

Evento	Heredado	Comentario
AutoSizeChanged	Control	Se produce cuando cambia el valor de la propiedad AutoSize.
BackColorChanged	Control	Se produce cuando cambia el valor de la propiedad BackColor
BackgroundImageChanged	Control	Se produce cuando cambia el valor de la propiedad BackgroundImage
BindingContextChanged	Control	Se produce cuando cambia el valor de la propiedad BindingContext
CausesValidationChanged	Control	Se produce cuando cambia el valor de la propiedad CausesValidation.
ChangeUICues	Control	Se produce cuando cambian las guías de la interfaz de usuario para el foco o el teclado.
ContextMenuChanged	Control	Se produce cuando cambia el valor de la propiedad ContextMenu.
ControlAdded	Control	Se produce cuando se añade un objeto a la colección Control.ControlCollection.
ControlRemoved	Control	Se produce cuando se elimina un objeto de la colección Control.ControlCollection.
CursorChanged	Control	Se produce cuando cambia el valor de la propiedad Cursor.
DrawItem		Cuando cambia la apariencia del control ListBox
Disposed	Control	Agrega un controlador de eventos para escuchar al evento Disposed en el componente.
DockChanged	Control	Se produce cuando cambia el valor de la propiedad Dock.
EnabledChanged	Control	Se produce cuando cambia el valor de la propiedad Enabled.
FontChanged	Control	Se produce cuando cambia el valor de la propiedad Font.
ForeColorChanged	Control	Se produce cuando cambia el valor de la propiedad ForeColor.

ImeModeChanged	Control	Se produce cuando cambia el valor de la propiedad ImeMode
LocationChanged	Control	Se produce cuando cambia el valor de la propiedad Location.
ParentChanged	Control	Se produce cuando cambia el valor de la propiedad Parent.
RightToLeftChanged	Control	Se produce cuando cambia el valor de la propiedad RightToLeft.
SelectedIndexChanged		Se produce cuando cambia el valor de la propiedad SelectedIndex.
StyleChanged	Control	Se produce cuando cambia el estilo del objeto.
SystemColorsChanged	Control	Se produce cuando cambia los colores del sistema.
TabIndexChanged	Control	Se produce cuando cambia el valor de la propiedad TabIndex.
TabStopChanged	Control	Se produce cuando cambia el valor de la propiedad TabStop.
TextAlignChanged		Se produce cuando cambia el valor de la propiedad TextAlign.
TextChanged	Control	Se produce cuando cambia el valor de la propiedad Text.
VisibleChanged	Control	Se produce cuando cambia el valor de la propiedad Visible.
<b>Teclado</b>		
KeyDown	Control	Se produce cuando cambia se usa el teclado con el objeto.
KeyUp	Control	Se produce cuando cambia se usa el teclado con el objeto.
KeyPress	Control	Se produce cuando cambia se usa el teclado con el objeto.
<b>Ratón</b>		
Click, DoubleClick	Control	Se produce cuando se hace clic con el ratón en el objeto.
MouseDown, MouseUp	Control	Se produce cuando el botón del ratón baja o sube.
MouseEnter	Control	Se produce cuando el puntero del mouse entra en el objeto.
MouseOver	Control	Se produce cuando el puntero del mouse esta encima del objeto.
MouseLeave	Control	Se produce cuando el puntero del mouse abandona el objeto.
MouseWheel	Control	Se produce cuando la rueda del mouse se mueve.
<b>Arrastrar y soltar</b>		
DragDrop	Control	Se produce cuando termina la acción de arrastrar y soltar.
DragEnter	Control	Se produce cuando se arrastra un objeto dentro de los límites del objeto.
DragLeave	Control	Se produce cuando se arrastra un objeto fuera de los límites del objeto.
DragOver	Control	Se produce cuando se arrastra un objeto sobre de los límites del objeto.
GiveFeedBack	Control	Se produce durante una operación de arrastre.
QueryContinueDrag	Control	Se produce durante una operación de arrastrar y soltar, y permite al origen de arrastre determinar si la operación de arrastrar y soltar tiene que cancelarse.
<b>Referentes al foco</b>		
Enter	Control	Se produce cuando se entra en el objeto.
GotFocus	Control	Se produce cuando el objeto toma el foco.
LostFocus	Control	Se produce cuando se pierde el foco.
Leave	Control	Se produce cuando el foco de entrada deja el control.
Validated	Control	Se produce cuando finaliza la validación del objeto.
Validating	Control	Se produce cuando se está validando el objeto.
<b>Otros</b>		
HandleCreated	Control	Se produce cuando se crea un identificador para el objeto.
HandleDestroyed	Control	Se produce cuando se elimina un identificador para el objeto.
HelpRequest	Control	Se produce cuando el usuario solicita ayuda sobre el objeto.
Invalidated	Control	Se produce cuando es necesario volver a dibujar el objeto de nuevo.
Layout	Control	Se produce cuando un objeto necesita volver a colocar sus objetos secundarios.

MeasureItem		Se produce cuando se crea un control ListBox dibujado por el propietario y se determina el tamaño de los elementos de la lista.
Move	Control	Se produce cuando cambia de ubicación el objeto.
Paint	Control	Se produce cuando se vuelve a dibujar el objeto.
Resize	Control	Se produce cuando cambia el tamaño del objeto.

#### 1.4.5 Específicos del Formulario, Form.

Evento	Heredado	Comentario
Activated		Se produce cuando cambia se activa el Form, por código o por el usuario.
Closed		Tiene lugar cuando el formulario está cerrado.
Closing		Tiene lugar cuando el formulario se cierra el formulario.
DeActivate		Tiene lugar cuando el formulario pierde el foco.
InputLenguajeChanged		Se produce después de cambiarse el idioma de entrada del formulario.
InputLenguajeChanging		Se produce después cuando el usuario intenta cambiar el idioma de entrada del formulario.
MaximizeBoundshanged		Se produce cuando cambia el valor de la propiedad MaximizedBounds.
MaximunSizeChanged		Se produce cuando cambia el valor de la propiedad MaximunSize
MenuStart		Se produce cuando el menú de un formulario recibe el foco.
MinimunSizeChanged		Se produce cuando cambia el valor de la propiedad ManmunSize

#### 1.4.6 Específicos de TextBox

Evento	Heredado	Comentario
AcceptsTabChanged	TextBoxBase	Se produce cuando cambia el valor de AcceptsTab
BorderStycleChanged	TextBoxBase	Se produce cuando cambia el valor de la propiedad BorderStyle
HideSelectionCHanged	TextBoxBase	Se produce cuando cambia el valor de la propiedad HideSelection
ModifiedChanged	TextBoxBase	Se produce cuando cambia el valor de la propiedad Modified.
ReadOnlyChanged	TextBoxBase	Se produce cuando cambia el valor de la propiedad ReadOnly.

#### 1.4.7 Específicos de ListBox, CheckedListBox, ComboBox

Evento	Heredado	Comentario
DataSourceChanged	ListControl	Se produce cuando cambia el valor de la propiedad DataSource.
DisplayMemberChanged	ListControl	Se produce cuando cambia el valor de la propiedad DisplayMember
DrawItem		Se produce cuando cambia el aspecto visual del objeto
DropDown		Se produce cuando se despliega la parte visible del ComboBox
DropDownStyleCHanged		Se produce cuando cambia el estilo de DropDownStyle
ItemCheck		Se produce cuando cambia de estado un elemento.
SelectedIndexChanged		Se produce cuando cambia el valor de SelectedIndex
SelectedValueChanged	ListControl	Se produce cuando cambia el valor de la propiedad SelectedValue.
ValueMemberChanged	ListControl	Se produce cuando cambia el valor de la propiedad ValueMember.

## 1.4.8 Específicos de DataGridView

Evento	Heredado	Comentario
AllowNavigationChanged		Se produce cuando cambia el valor de la propiedad AllowNavigation
BackButtonClick		Se produce cuando se hace clic en el botón de retroceso de una tabla secundaria.
CaptionVisibleChanged		Se produce cuando cambia la propiedad CaptionVisible
DataSourceChanged		Se produce cuando cambia la propiedad DataSource
FlatModeChanged		Se produce cuando cambia la propiedad FlatMode
ParentRowsLabelStyleChanged		Se produce cuando cambia el estilo de la etiqueta de la fila primaria
ParentRowsVisibleChanged		Se produce cuando cambia la propiedad ParentRowsVisible
Scroll		Se produce cuando el usuario desplaza el control System.Windows.Forms.DataGridView.
ShowParentDetailsButtonClick		Se produce cuando se hace clic en el botón ShowParentDetails.
<b>Protegido</b>		
RowHeaderClick		Se produce cuando se hace clic en un encabezado de fila

## 1.5 Métodos.

Los métodos han cambiado también en su nombre y en el número de posibilidades que abarcan. Están compuestos por instrucciones ejecutables de un programa.

Tienen una lista de parámetros formales (que puede estar vacía), un valor devuelto (o ningún valor devuelto) y ser compartidos o no compartidos. El acceso a los métodos compartidos se obtiene a través de la clase. El acceso a los métodos no compartidos, también denominados métodos de instancias, se obtiene a través de instancias de la clase.

En el ejemplo siguiente se muestra una clase Stack con varios métodos compartidos (Clone y Flip) y varios métodos de instancias (Push, Pop y ToString).

```
Imports System
Public Class Stack

    Public Shared Function Clone(s As Stack) As Stack
    End Function

    Public Shared Function Flip(s As Stack) As Stack
    End Function

    Public Function Pop() As Object
    End Function

    Public Sub Push(o As Object)
    End Sub

    Public Overrides Function ToString() As String
    End Function
End Class

Module Test

    Sub Main()
        Dim s As New Stack()
        Dim i As Integer

        While i < 10
            s.Push(i)
        End While
        Dim flipped As Stack = Stack.Flip(s)
    End Sub
End Module
```

```

Dim cloned As Stack = Stack.Clone(s)
Console.WriteLine(("Original stack: " & s.ToString()))
Console.WriteLine(("Flipped stack: " & flipped.ToString()))
Console.WriteLine(("Cloned stack: " & cloned.ToString()))
End Sub
End Module

```

Los métodos se pueden sobrecargar, lo que significa que varios métodos pueden tener el mismo nombre siempre que tengan firmas únicas.

La firma de un método se compone del nombre del método y del número y los tipos de sus parámetros formales. La firma de un método no incluye específicamente el tipo de valor devuelto o los modificadores del parámetro.

En el siguiente ejemplo se muestra una clase con un número F de métodos:

```

Module Test
Sub F()
    Console.WriteLine("F()")
End Sub

Sub F(o As Object)
    Console.WriteLine("F(Object)")
End Sub

Sub F(ByVal Value As Integer)
    Console.WriteLine("F(Integer)")
End Sub

Sub F(a As Integer, b As Integer)
    Console.WriteLine("F(Integer, Integer)")
End Sub

Sub F(values() As Integer)
    Console.WriteLine("F(Integer[])")
End Sub

Sub Main()
    F()
    F(1)
    F(CType(1, Object))
    F(1, 2)
    F(New Integer() {1, 2, 3})
End Sub
End Module

```

El resultado de este programa es:

```

F()
F(Integer)
F(Object)
F(Integer, Integer)
F(Integer())

```

## 1.5.1 Desaparecen

Estos son los métodos controlados que desaparecen:

LinkExecute  
LinkPoke  
LinkRequest  
LinkSend

## 1.5.2 Genéricos

<b>Método</b>	<b>Anterior</b>	<b>Comentario</b>
BringToFront	ZOrder	Coloca el control al principio del orden Z.
SendToBack	ZOrder	Coloca el control al final del orden Z.
Contains		Obtiene un valor que indica si el control especificado es un control secundario del control.
CreateControl		Obliga a que se cree el control, incluidos el identificador y los controles secundarios.
CreateGraphics		Crea un objeto Graphics para el formulario.
CreateObjRef		Crea un objeto que contiene toda la información relevante necesaria para generar un proxy utilizado para comunicarse con un objeto remoto.
Dispose		Sobrecargado. Libera los recursos utilizados por Dispose.
DoDragDrop		Inicia una operación de arrastrar y soltar.
FindForm		Recupera el formulario en el que se encuentra el control.
Focus	SetFocus	Establece el foco en el objeto
GetContainerControl		Devuelve el siguiente ContainerControl (en sentido ascendente) de la cadena de controles principales del control.
GetChildAtPoint		Recupera el control secundario ubicado en las coordenadas especificadas.
GetNextControl		Recupera el siguiente control, hacia delante o hacia atrás, en el orden de tabulación de controles secundarios.
GetType		Obtiene el objeto tipo del control actual.
Hide		Oculto el objeto.
Invalidate		Sobrecargado. Invalida una región específica del control y hace que se envíe un mensaje de dibujo al control.
PointToClient		Calcula la ubicación del punto especificado de la pantalla, en coordenadas de cliente.
PointToScreen		Calcula la ubicación del punto especificado de cliente en coordenadas de pantalla.
RectangleToClient	Line	Calcula el tamaño y la ubicación del rectángulo de pantalla especificado, en coordenadas de cliente.
RectangleToScreen	Line	Calcula el tamaño y la ubicación del rectángulo de pantalla especificado, en coordenadas de pantalla
ResetBackColor		Restablece el valor predeterminado de la propiedad BackColor
ResetBindings		Restablece el valor predeterminado de la propiedad Bindings
ResetCursor		Restablece el valor predeterminado de la propiedad Cursor
ResetFont		Restablece el valor predeterminado de la propiedad Font
ResetForeColor		Restablece el valor predeterminado de la propiedad ForeColor
ResetRightToLeft		Restablece el valor predeterminado de la propiedad ResetRightToLeft
ResetText		Restablece el valor predeterminado de la propiedad Text
Refresh		Obliga al control a invalidar su área cliente y, acto seguido, obliga a que vuelva a dibujarse el control y sus controles secundarios.
Scale	Scale	Sobrecargado. Ajusta la escala del control y de todos los controles secundarios.
SelectNextControl	SetFocus	Activa el siguiente objeto.
Update	DoEvents	Actualiza el formulario.

## Protegidos

DrawImage  
GetService

GetStyle

GetTopLevel  
InvokeGotFous  
InvokeLostFocus  
InvokeOnClick  
InvokePaint  
InvokePaintBackground  
IsInputChar

IsInputKey

OnAppearanceChanged  
OnAutoSizeChanged  
OnBackColorChanged  
OnBackGroundImageChanged  
OnCausesValidationChanged  
OnClick  
OnControlAdded  
OnControlRemoved  
OnCursorChanged  
OnDockChanged  
OnDoubleClick  
OnDragDrop  
OnDragEnter  
OnDragLeave  
OnDragOver  
OnEnter  
OnForeColorChanged  
OnFontChanged  
OnGotFocus  
OnInvalidated  
OnKeyDown  
OnKeyPress  
OnKeyUp  
OnLayout  
OnLeave  
OnLocationChanged  
OnLostFocus  
OnMouseDown  
OnMouseEnter  
OnMouseHover  
OnMouseLeave  
OnMouseMove  
OnMouseUp  
OnMouseWheel  
OnMove  
OnNotifyMessage  
OnPaintBackGround  
OnParentBackGroundChanged

OnParentEnabledChanged

## Comentarios

Dibuja la imagen dentro de los límites especificados.  
Devuelve un objeto que representa el servicio suministrado por Component o por Container.  
Recupera el valor del bit de estilo de control especificado para el control.  
Determina si el objeto es de nivel superior.  
Provoca el evento GotFocus  
Provoca el evento LostFocus  
Provoca el evento Click  
Provoca el evento Paint  
Provoca el evento PaintBackground  
Determina si un carácter es un carácter de entrada que el control reconoce.  
Determina si la tecla especificada es una tecla de entrada normal o una tecla especial que requiere preprocesamiento.

Provoca el evento AppearanceChanged  
Provoca el evento AutoSizeChanged  
Provoca el evento BackColorChanged  
Provoca el evento BackGroundImageChanged  
Provoca el evento CausesValidationChanged  
Provoca el evento Click  
Provoca el evento ControlAdded  
Provoca el evento ControlRemoved  
Provoca el evento CursorChanged  
Provoca el evento DockChanged  
Provoca el evento DoubleClick  
Provoca el evento DragDrop  
Provoca el evento DragEnter  
Provoca el evento DragLeave  
Provoca el evento DragOver  
Provoca el evento Enter  
Provoca el evento ForeColorChanged  
Provoca el evento FontChanged  
Provoca el evento GotFocus  
Provoca el evento Invalidated  
Provoca el evento KeyDown  
Provoca el evento KeyPress  
Provoca el evento KeyUp  
Provoca el evento Layout  
Provoca el evento Leave  
Provoca el evento LocationChanged  
Provoca el evento LostFocus  
Provoca el evento MouseDown  
Provoca el evento MouseEnter  
Provoca el evento MouseHover  
Provoca el evento MouseLeave  
Provoca el evento MouseMove  
Provoca el evento MouseUp  
Provoca el evento MouseWheel  
Provoca el evento Move  
Notifica al control los mensajes de Windows.  
Pinta el fondo del control.  
Provoca el evento BackGroundImageChanged cuando cambia el valor de la propiedad BackGroundImage del contenedor del control.  
Provoca el evento EnabledChanged cuando cambia el valor de la propiedad Enabled del contenedor del control.

OnParentFontChanged	Provoca el evento FontChanged cuando cambia el valor de la propiedad Font del contenedor del control.
OnParentForeColorChanged	Provoca el evento ForeColorChanged cuando cambia el valor de la propiedad ForeColor del contenedor del control.
OnParentVisibleChanged	Provoca el evento VisibleChanged cuando cambia el valor de la propiedad Visible del contenedor del control.
OnQueryContinueDrag	Provoca el evento QueryContinueDrag
OnResize	Provoca el evento Resize
OnTextChanged	Provoca el evento TextChanged
OnValidated	Provoca el evento Validated
OnValidating	Provoca el evento Validating
ProcessCmdKey	Procesa una tecla de comando.
ProcessDialogKey	Procesa una tecla de cuadro de diálogo.
ProcessKeyEventArgs	Procesa un mensaje de tecla y genera los eventos de control correspondientes.
ProcessKeyMessage	Procesa un mensaje de teclado.
ProcessKeyPreview	Muestra una vista preliminar de un mensaje del teclado.
Select	Activa el objeto.
SetStyle	Establece el bit de estilo especificado en el valor especificado.
UpdateStyles	Obliga a que los estilos asignados vuelvan a aplicarse al objeto.
UpdateZOrder	Actualiza el objeto en el eje Z

### 1.5.3 Formulario.

Método	Anterior	Comentario
Activate		Activa el formulario y le otorga el foco.
AddOwnedForm		Agrega un formulario con propietario al formulario.
BeginInvoke		Sobrecargado. Ejecuta un delegado de forma asincrónica en el subproceso donde se creó el identificador subyacente del control.
Close	UnLoad	Cierra el formulario
Contains		Obtiene un valor que indica si el control especificado es un control secundario del control.
CreateControl		Obliga a que se cree el control, incluidos el identificador y los controles secundarios.
Hide		Oculto el formulario.
Show	Show	Muestra el formulario.

### 1.5.4 Button, CommandButton.

Son todos estándar

### 1.5.5 TextBox.

Método	Anterior	Comentario
AppendText		Anexa texto al texto actual del cuadro de texto.
Clear	Clear	Borra todo el texto del control de cuadro de texto.
ClearUndo		Borra del búfer Deshacer del cuadro de texto la información sobre la última operación realizada.
Copy		Copia la selección actual del cuadro de texto en el Portapapeles.
CreateControl		Obliga a que se cree el control, incluidos el identificador y los controles secundarios.
Cut		Mueve la selección actual del cuadro de texto al Portapapeles.
Paste		Reemplaza la selección actual del cuadro de texto con el contenido del Portapapeles.
PreProcessMessage		Preprocesa los mensajes de entrada en el bucle de mensajes antes de enviarlos.
Select		Selecciona texto en el objeto.
SelectAll		Selecciona todo el texto en el objeto.

ToString	Devuelve un objeto String que representa al objeto Object actual.
Undo	Deshace la última operación de edición del cuadro de texto.

### 1.5.6 Label

Son todos estándar.

### 1.5.7 ListBox

Método	Anterior	Comentario
BeginUpdate		Mantiene el rendimiento mientras se agregan elementos al control ListBox de uno en uno al impedir que se dibuje el control hasta que se llame al método EndUpdate.
ClearSelected		Anula la selección de todos los elementos del control ListBox.
EndUpdate		Reanuda el dibujo del control ListBox después de haber sido suspendido por el método BeginUpdate.
FindString		Sobrecargado. Busca el primer elemento del control ListBox que comience por la cadena especificada.
GetItemHeight		Devuelve el alto de un elemento del control ListBox.
GetSelected		Devuelve un valor que indica si el elemento especificado está seleccionado.
OnDataSourceChanged		Provoca el evento DataSourceChanged
OnDisplayMemberChanged		Provoca el evento DisplayMemberChanged
OnMeasureItem		Provoca el evento MeasureItem.
OnValueMemberChanged		Provoca el evento ValueMemberChaned
RefreshItem		Sincroniza de nuevo los datos del objeto en el índice especificado, con el contenido del origen de datos.
Sort		Ordena alfabéticamente los elementos del control ListBox .

### 1.5.8 ComboBox

Método	Anterior	Comentario
FindString		Sobrecargado. Busca el primer elemento del ComboBox que comienza con la cadena especificada.
FindStringExact		Sobrecargado. Busca el elemento que coincide exactamente con la cadena especificada.
GetItemHeight		Devuelve el alto de un elemento del ComboBox .
GetItemText		Devuelve la representación de texto del elemento especificado.
Select		Selecciona texto en el objeto.
SelectAll		Selecciona todo el texto en el objeto.
OnDataSourceChanged		Provoca el evento DataSourceChanged
OnDisplayMemberChanged		Provoca el evento DisplayMemberChanged
OnDrawItem		Provoca el evento DrawItem
OnDropDraw		Provoca el evento DropDraw
OnDropDownStyleChanged		Provoca el evento DropDownStyleChanged
OnSelectionChangeOmmited		Provoca el evento SelectionChangCommitted
OnSelectionIndexChanged		Provoca el evento SelectionIndexChanged
OnValueMemberChanged		Provoca el evento ValueMemberChaned
RefreshItem		Sincroniza de nuevo los datos del objeto en el índice especificado, con el contenido del origen de datos.

### 1.5.9 CheckBox

Método	Anterior	Comentario
OnCheckedChanged		Provoca el evento CheckedChanged
OnCheckedStateChanged		Provoca el evento CheckedStateChanged

