



Interacción Persona Ordenador

Visual Basic

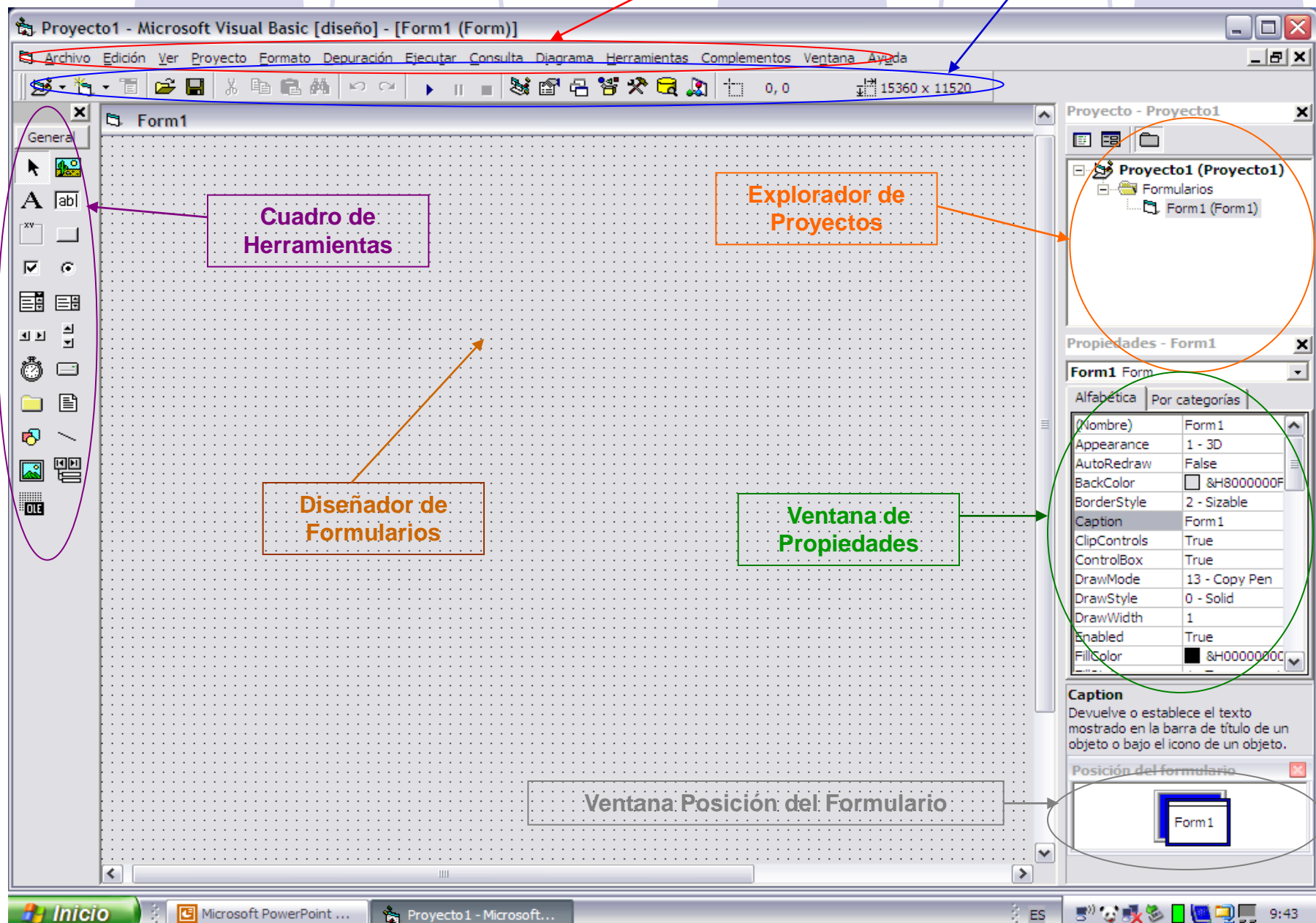
Tema 2

Elementos del **E**ntorno **I**ntegrado de
Desarrollo (IDE)

Visual Basic

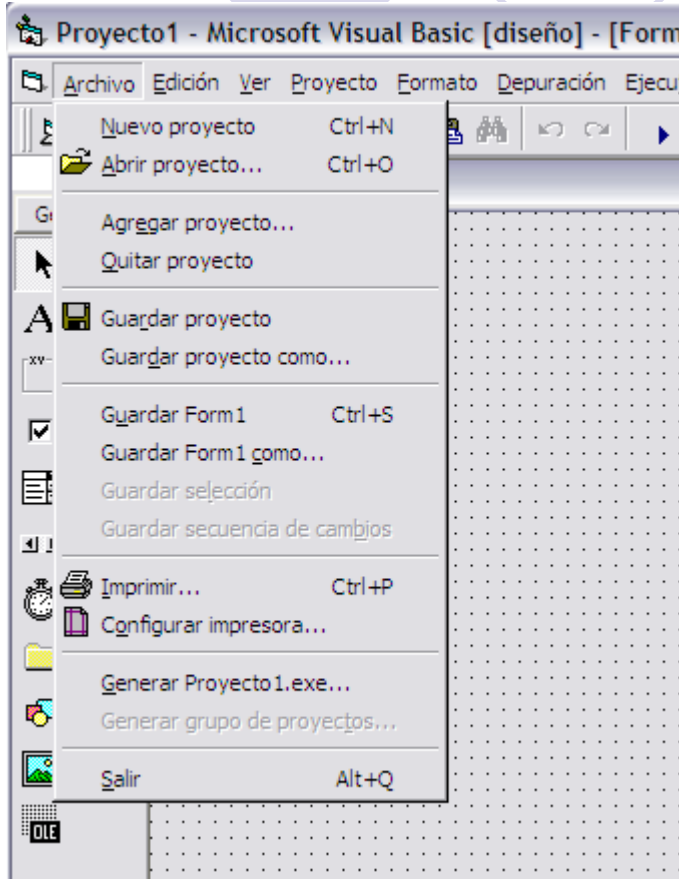
Barra de Menú

Barra de Herramientas



Visual Basic

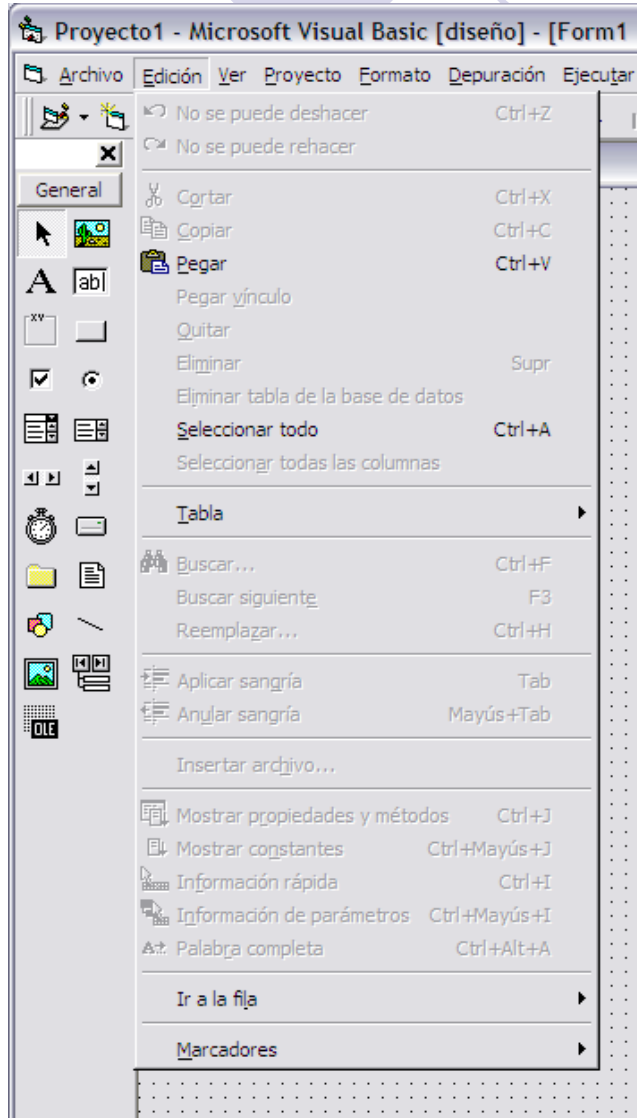
Barra de Menú - Archivo



- Aquí encontraremos las opciones necesarias para crear un nuevo proyecto o cargar uno ya existente, guardar un archivo individual, guardar todo el proyecto, añadir y eliminar elementos del proyecto, imprimir los formularios o el código de la aplicación y generar un ejecutable. Al final de esta ventana siempre encontraremos una lista con los últimos archivos sobre los que hemos trabajado, permitiéndonos así un acceso rápido a ellos.

Visual Basic

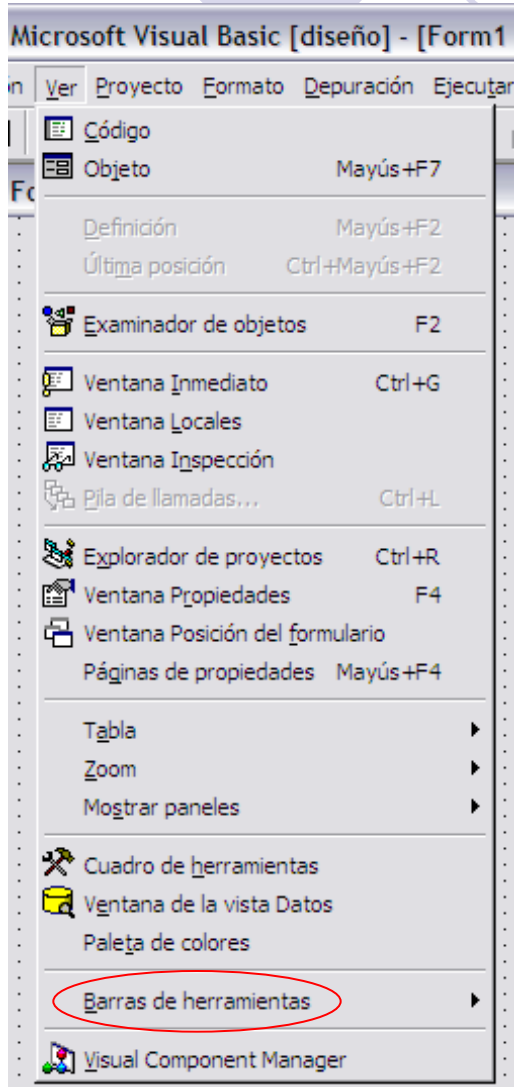
Barra de Menú - Edición



- En el menú de edición además de las opciones típicas de deshacer, rehacer, copiar, pegar, búsqueda y sustitución, encontraremos otras que nos permitirán controlar la “*indentación*” del código y controlar la posición de los controles dentro del formulario.

Visual Basic

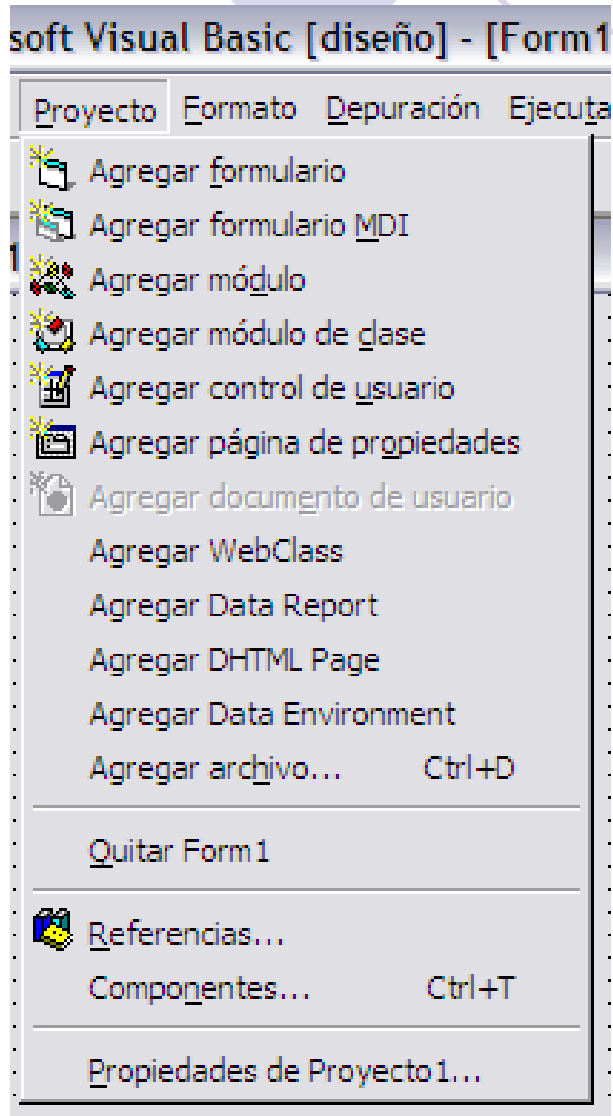
Barra de Menú - Ver



- Dentro de este menú encontraremos todas aquellas opciones que nos permiten seleccionar la información que deseamos ver en un momento dado. El código asociado a un control, el formulario, el proyecto, la barra de herramientas o la paleta de colores, la lista de propiedades o la ventana de depuración, etc. También encontraremos una opción, **Barra de herramientas**, con la que podremos eliminar o visualizar la barra de botones de Visual Basic.

Visual Basic

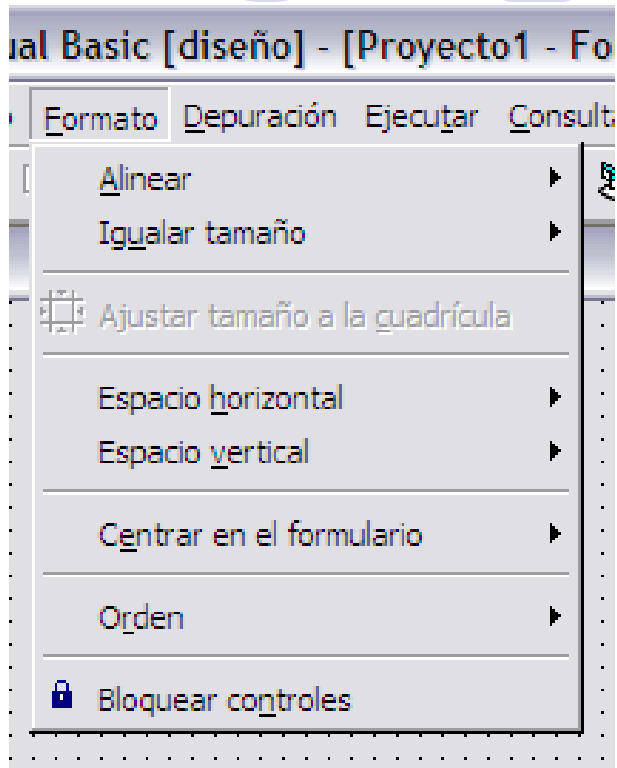
Barra de Menú - **Proyecto**



- Las opciones que encontraremos aquí nos permitirán insertar en nuestro proyecto nuevos formularios, si es que va a tener más de uno, módulos de código o definiciones de clases, nuevas funciones y procedimientos e incluso agregar archivos de texto internos.

Visual Basic

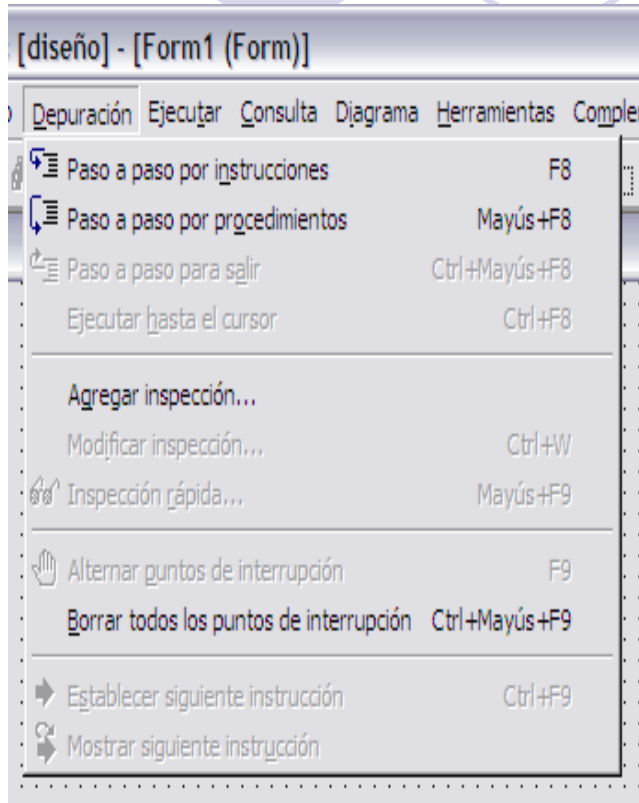
Barra de Menú - **Formato**



- Nos suministra una serie de opciones para centrar, ajustar y distribuir los distintos controles en y dentro del formulario sobre el que se está trabajando.

Visual Basic

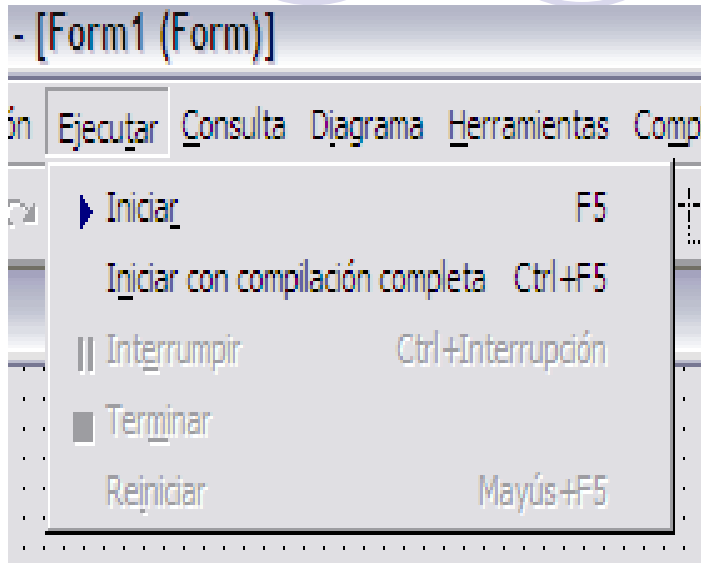
Barra de Menú - Depuración



- Este menú contiene todas las opciones necesarias para la depuración de un programa. Podemos ejecutar el programa paso a paso por instrucciones o paso a paso por procedimientos, siendo posible establecer puntos de ruptura en el código.

Visual Basic

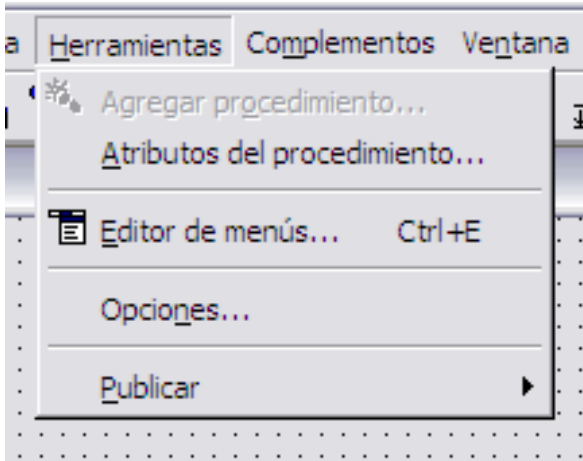
Barra de Menú - **Ejecutar**



- Este menú contiene todas las opciones relacionadas con la ejecución del programa desde dentro del entorno Visual Basic. La ejecución se puede detener y reanudar según nuestras necesidades.

Visual Basic

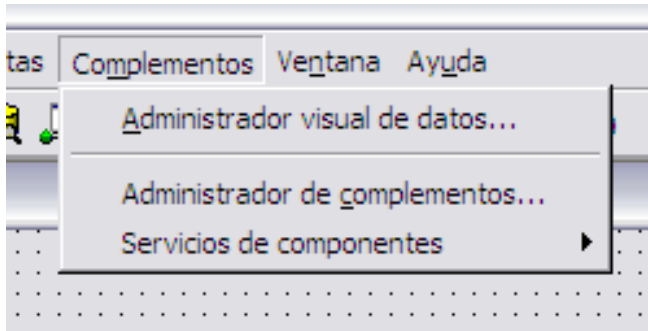
Barra de Menú - Herramientas



- Aquí encontraremos las herramientas necesarias que nos permitirán diseñar menús, seleccionar los controles u objetos que deben aparecer en nuestra caja de herramientas, establecer las referencias que deseamos tener activas y establecer las opciones de trabajo generales y de proyecto.

Visual Basic

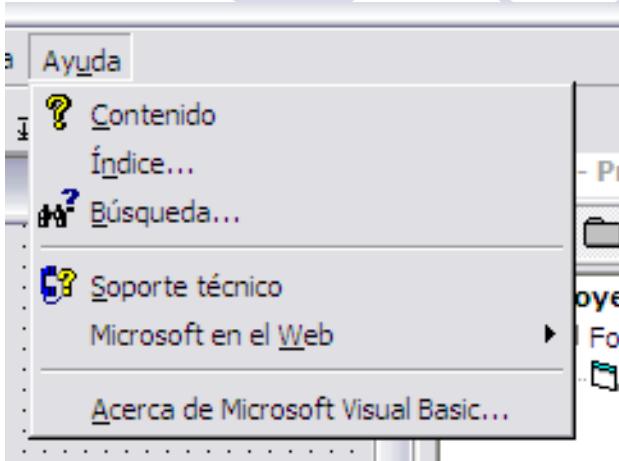
Barra de Menú - Complementos



- Visual Basic está abierto a la utilización de otras herramientas no incorporadas en el propio entorno, y para ello existe este menú. En él encontraremos opciones externas, tales como el Administrador visual de datos, o el Administrador de Complementos y la de Servicios de componentes.

Visual Basic

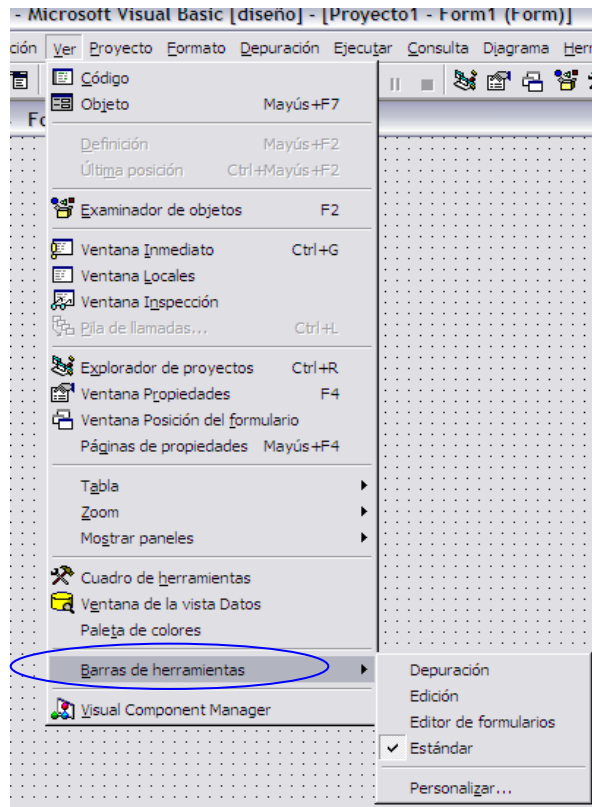
Barra de Menú - Ayuda



- Es éste un menú existente en la práctica totalidad de las aplicaciones, cuya finalidad es ayudarnos en cualquier momento en que tengamos una duda (MSDN).

Visual Basic

Barra de Herramientas

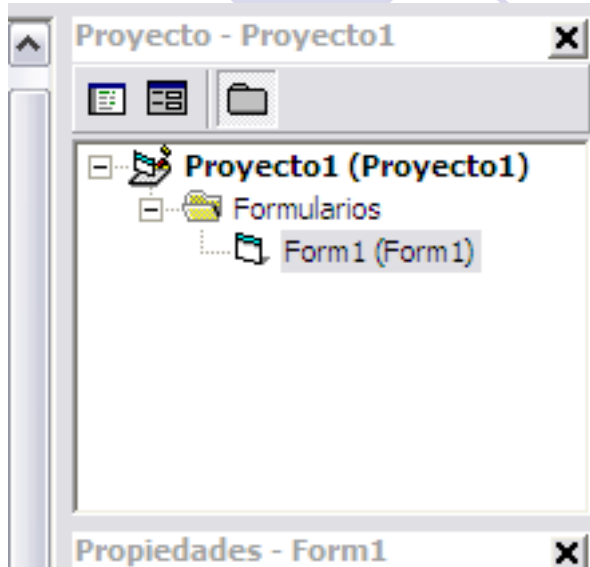


- Proporciona un rápido acceso a los comandos usados normalmente en el entorno de programación. De forma predeterminada, al iniciar Visual Basic se presenta la barra de herramientas **Estándar** (para modificarla → Menú **Ver, Barra de Herramientas**).
- La Barra de herramientas puede “flotar” si se selecciona la barra vertical del borde izquierdo y se arrastra fuera de la barra de menús.



Visual Basic

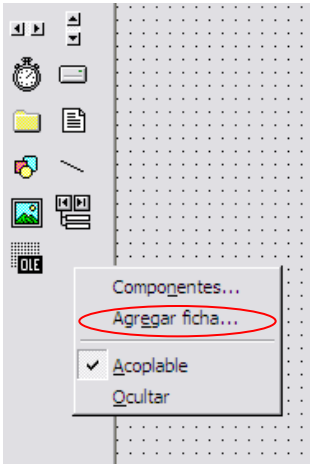
Explorador de Proyectos



- Enumera los formularios y módulos del proyecto actual. Un **Proyecto** es la colección de archivos que se usa para generar una aplicación.

Visual Basic

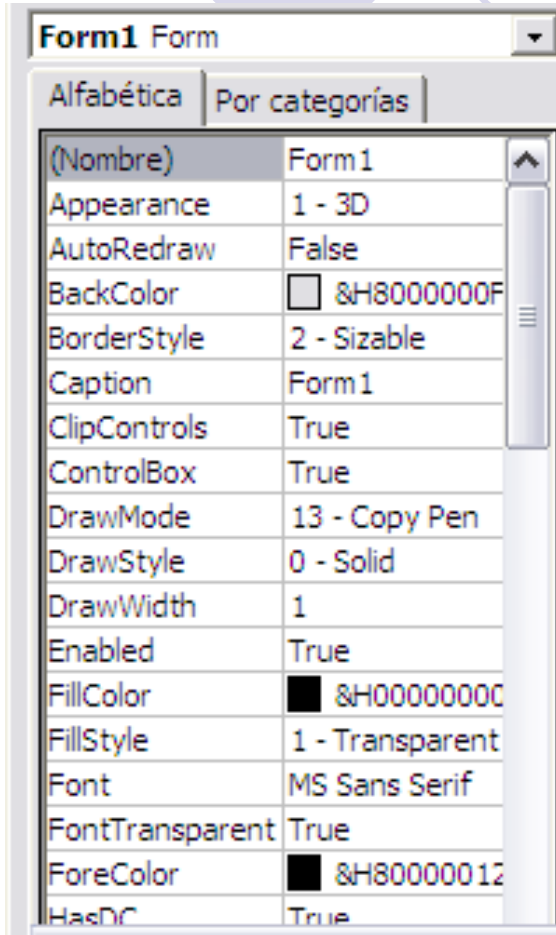
Cuadro de Herramientas



- Proporciona un conjunto de herramientas que podemos utilizar durante el diseño de la aplicación para colocar controles en un formulario.
- Podemos crear nuestro propio diseño personalizado seleccionando ***Agregar ficha*** en el menú contextual (menú que surge después de colocarnos con el ratón sobre el cuadro de herramientas y pulsar el botón auxiliar (derecho) de éste).

Visual Basic

Ventana de Propiedades



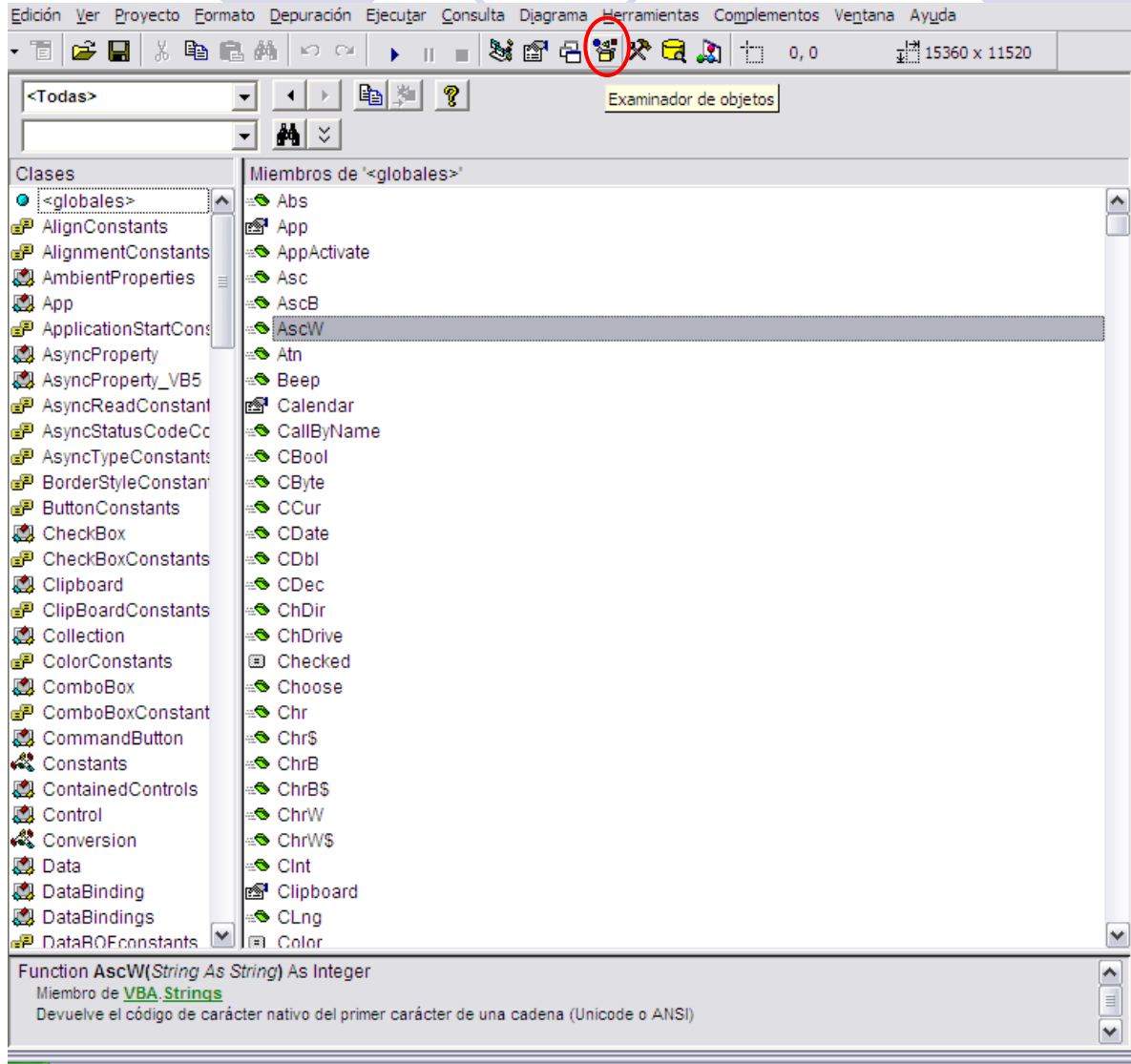
The screenshot shows the Visual Basic Properties Window for a form named 'Form1'. The window has two tabs: 'Alfabética' (selected) and 'Por categorías'. The properties are listed in a table with two columns: '(Nombre)' and the property value. The properties shown are:

(Nombre)	Form1
Appearance	1 - 3D
AutoRedraw	False
BackColor	<input type="checkbox"/> &H8000000F
BorderStyle	2 - Sizable
Caption	Form1
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid
DrawWidth	1
Enabled	True
FillColor	<input checked="" type="checkbox"/> &H00000000
FillStyle	1 - Transparent
Font	MS Sans Serif
FontTransparent	True
ForeColor	<input checked="" type="checkbox"/> &H80000012
HasDC	True

- Enumera los valores de las propiedades de control o formulario seleccionado. Una ***Propiedad*** es una característica de un objeto, con su tamaño, título o color.

Visual Basic

Examinador de Objetos



- Enumera los objetos disponibles que podemos utilizar en nuestro proyecto y nos proporciona una manera rápida de desplazarnos a través del código. Lo podemos utilizar también para explorar objetos en VB y otros, ver que métodos y propiedades están disponibles en esos objetos, y pegar código de procedimientos en nuestra aplicación

Visual Basic

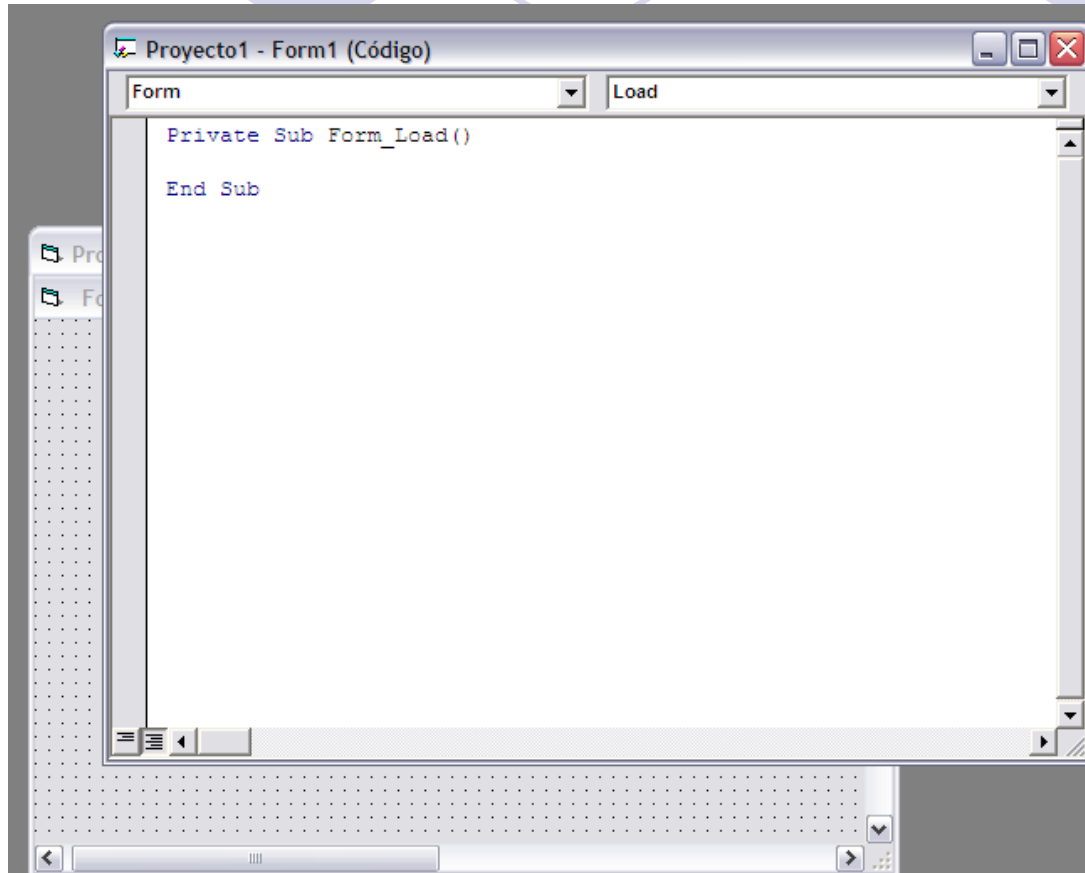
Diseñador de Formularios



- Funciona como una ventana en la que se personaliza el diseño de la interfaz de nuestra aplicación. Podemos agregar controles, gráficos e imágenes a nuestro formulario para crear la apariencia que deseemos. Cada formulario de la aplicación tiene su propia ventana “*Diseñador de Formulario*”.

Visual Basic

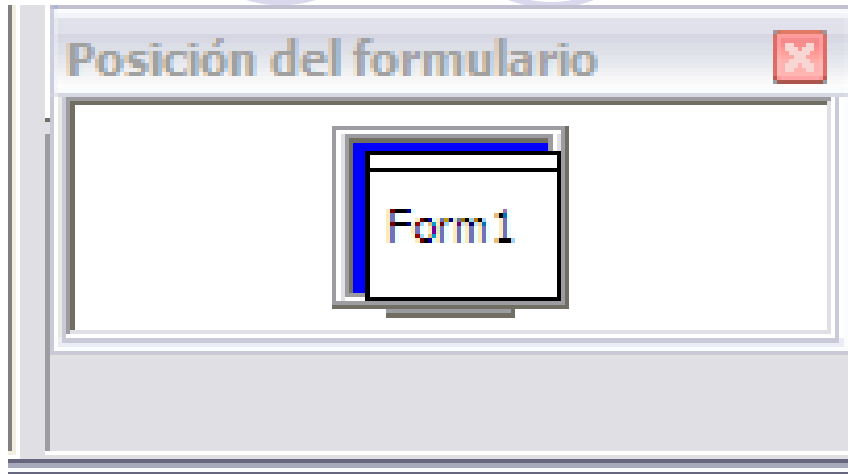
Ventana Editor de Código



- Funciona como un editor para escribir el código de la aplicación. Se crea una “*Ventana Editor de Código*” diferente para cada formulario o módulo de código de la aplicación.

Visual Basic

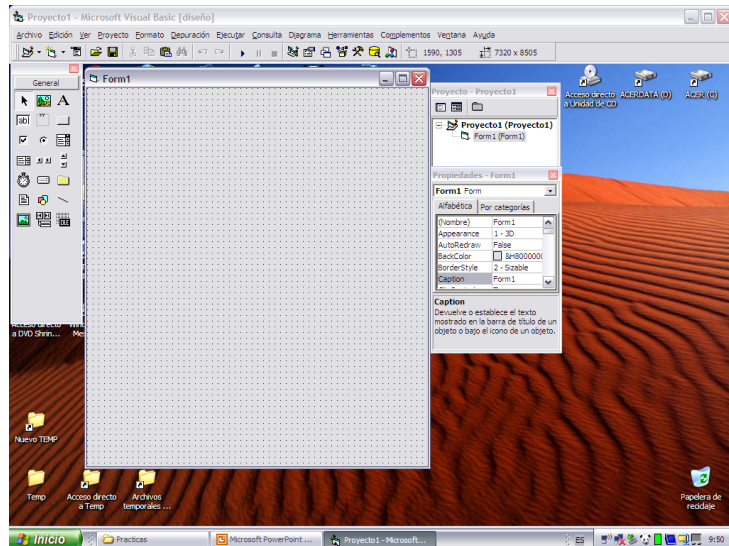
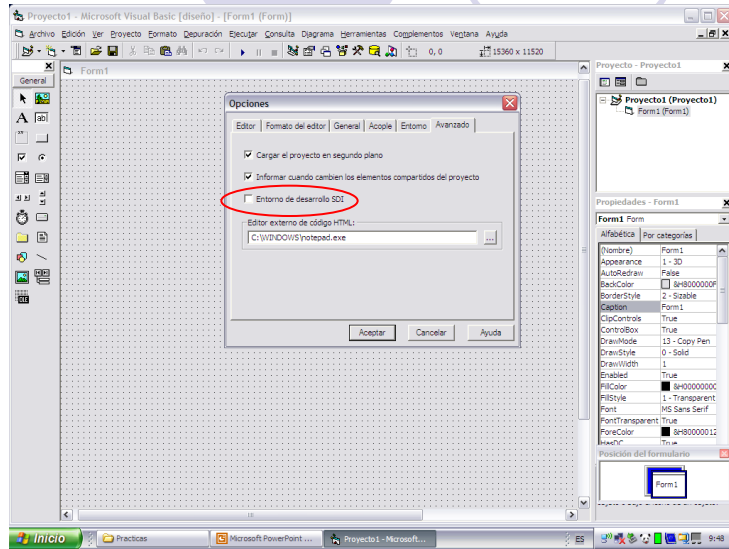
Ventana Posición del Formulario



- Nos permite colocar los formularios de nuestra aplicación utilizando una pequeña representación gráfica de la pantalla.

Visual Basic

Interfaz SDI o MDI



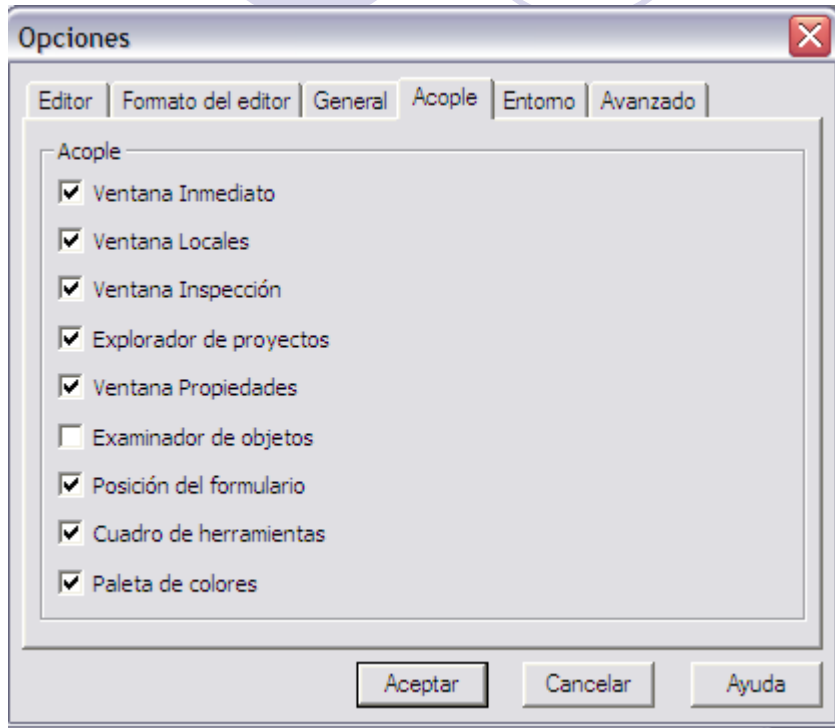
- Con la opción MDI, todas las ventanas del IDE están contenidas en una única ventana primaria de tamaño ajustable.

- Con la opción SDI, todas las ventanas del IDE se pueden mover libremente por cualquier lugar de la pantalla.

(Herramientas → Opciones → Avanzado → Entorno de desarrollo SDI)

Visual Basic

Acople



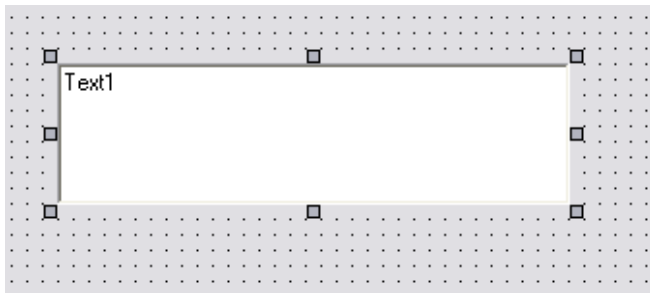
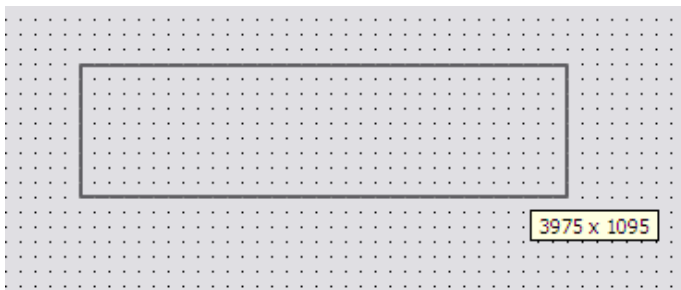
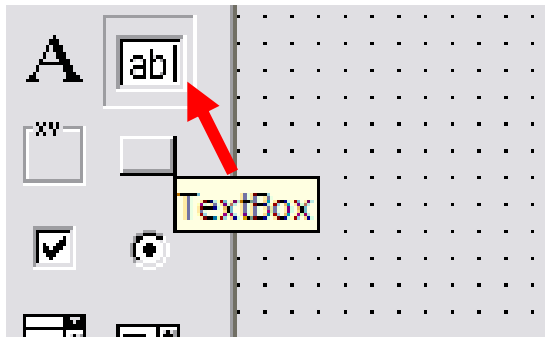
- Muchas de las ventanas del IDE se pueden “acoplar” o conectar a otra o al borde de la pantalla.
- Con la opción MDI las ventanas pueden acoplarse en cualquier lado de la ventana primaria.
- Con la opción SDI sólo se pueden acoplar debajo de la barra de menús.

(Herramientas → Opciones → Acople)

Visual Basic

Ejercicio 1

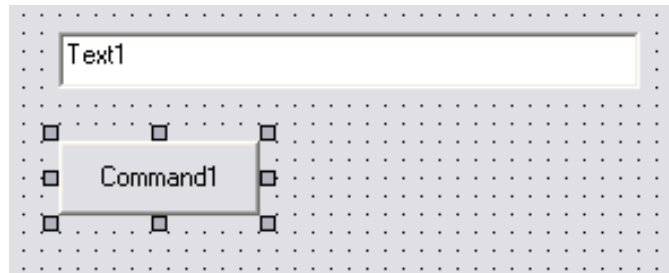
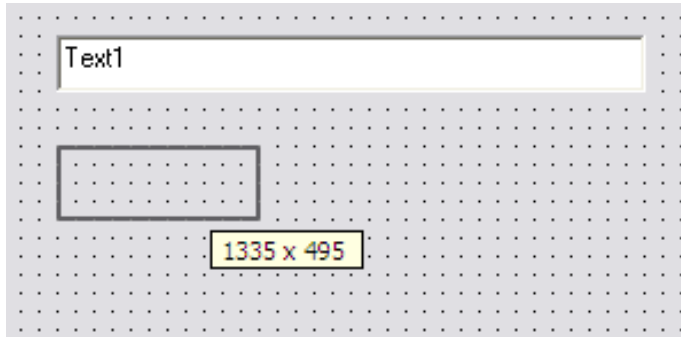
- Consiste en un cuadro de texto (**TextBox**) y un botón de comando (**CommandButton**) Cuando hagamos clic en el botón de comando aparecerá el mensaje en el cuadro e texto “*Buenos Días Señor@s*”.



- Hacemos clic sobre el control “*cuadro de texto*”.
- Nos vamos al formulario donde aparecerá una cruz “+”. Colocamos la cruz donde queramos que aparezca la esquina superior izquierda del “*cuadro de texto*”. Arrastraremos la cruz (pulsando el botón derecho del ratón y sin soltarlo) hasta obtener el tamaño deseado.
- Al soltar el botón del ratón, el “*cuadro de texto*” aparecerá en pantalla.
- (También con “*doble clic*”)
- (Cambiar tamaño: MAY + Teclas de Dirección)
- (Bloquear Controles: Formato → Bloquear controles)

Visual Basic

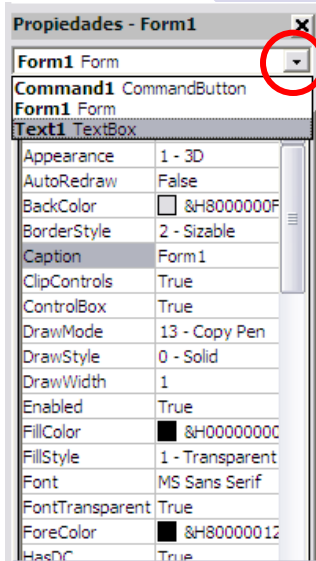
Ejercicio 1



- Hacemos clic sobre el control “*botón de comando*”.
- Nos vamos al formulario donde aparecerá una cruz “+”. Colocamos la cruz donde queramos que aparezca la esquina superior izquierda del “*botón de comando*”. Arrastraremos la cruz (pulsando el botón derecho del ratón y sin soltarlo) hasta obtener el tamaño deseado.
- Al soltar el botón del ratón, el “*botón de comando*” aparecerá en pantalla.
- (También con “*doble clic*”)

Visual Basic

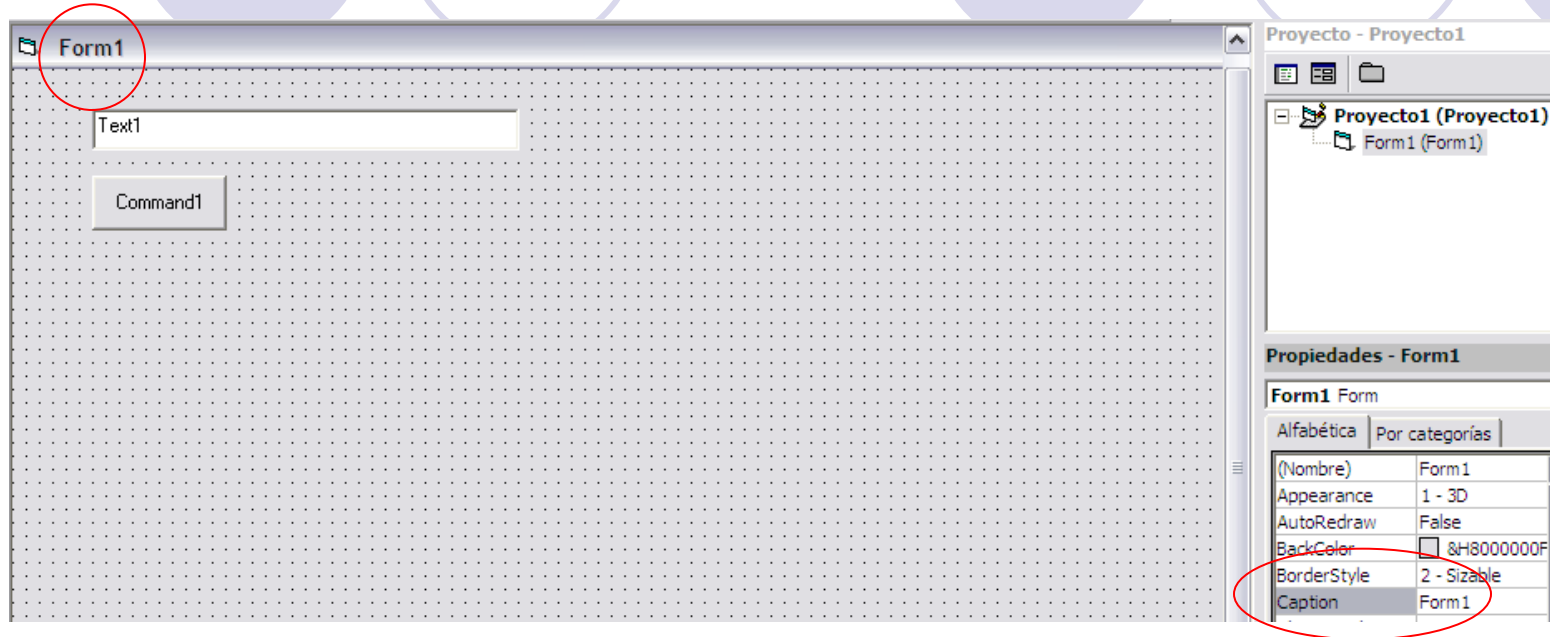
Ejercicio 1



- En la Ventana de Propiedades se pueden cambiar las propiedades de los objetos.
- Podemos ver los objetos que tiene nuestro formulario pulsando la flecha que hay a la derecha del Cuadro de Objetos.
- Nuestro ejercicio tiene 3 objetos:
 - El propio Formulario, que se llama internamente **Form1**.
 - El Cuadro de Textos, que se llama internamente **Text1**.
 - El Botón de Comando, que se llama internamente **Command1**.

Visual Basic

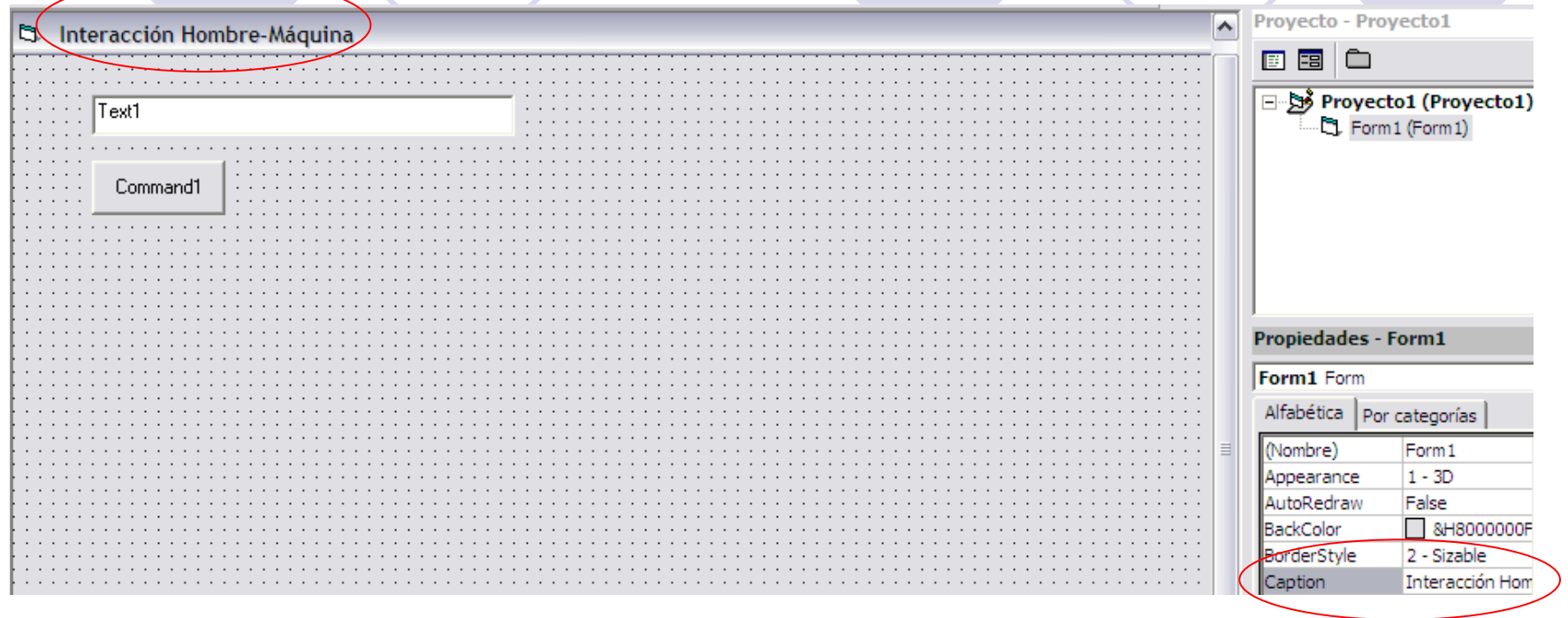
Ejercicio 1



- La propiedad **Caption** de los controles nos permite cambiar el nombre con el que aparecen los objetos en el formulario, **no** el nombre interno.
- Vamos a cambiar la propiedad *Caption* del formulario, que originalmente puede ser *Form1*, por el de *Interacción Hombre-Máquina*.

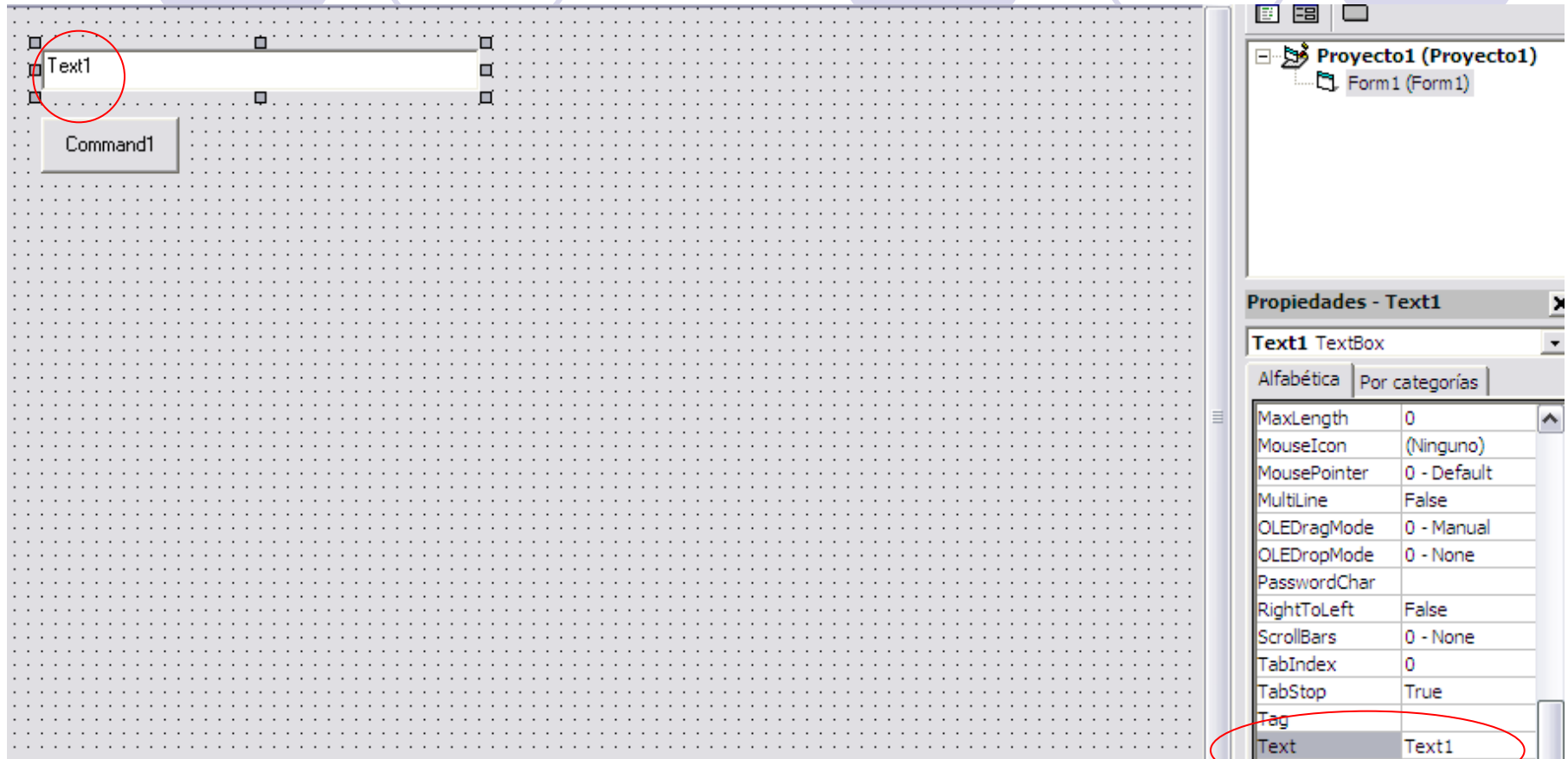
Visual Basic

Ejercicio 1



Visual Basic

Ejercicio 1



- Vamos a cambiar la propiedad **Text** del control Cuadro de Textos poniéndolo a blancos. Esta propiedad nos permite visualizar un texto en el Cuadro de Textos al comienzo de la ejecución de la aplicación.

Visual Basic

Ejercicio 1

The screenshot shows the Visual Basic IDE. On the left, a form is displayed with a grid background. It contains a Command1 button and a text box (Text1) which is circled in red. On the right, the Properties window is open, showing the properties for Text1. The 'Text' property is circled in red.

Propiedades - Text1

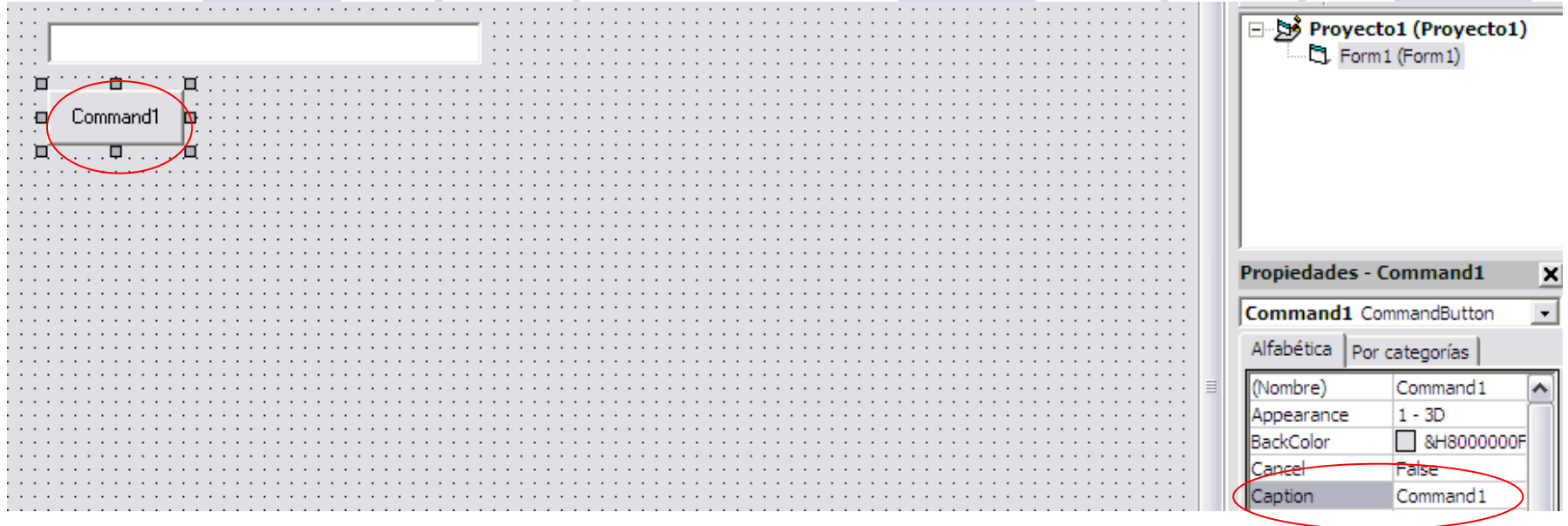
Text1 TextBox

Alfabetica | Por categorias

MaxLength	0
MouseIcon	(Ninguno)
MousePointer	0 - Default
MultiLine	False
OLEDragMode	0 - Manual
OLEDropMode	0 - None
PasswordChar	
RightToLeft	False
ScrollBars	0 - None
TabIndex	0
TabStop	True
Tag	
Text	

Visual Basic

Ejercicio 1



- Vamos a cambiar la propiedad **Caption** del control Botón de Comando que inicialmente tiene *Command1* por el literal **Aceptar**.

Visual Basic

Ejercicio 1

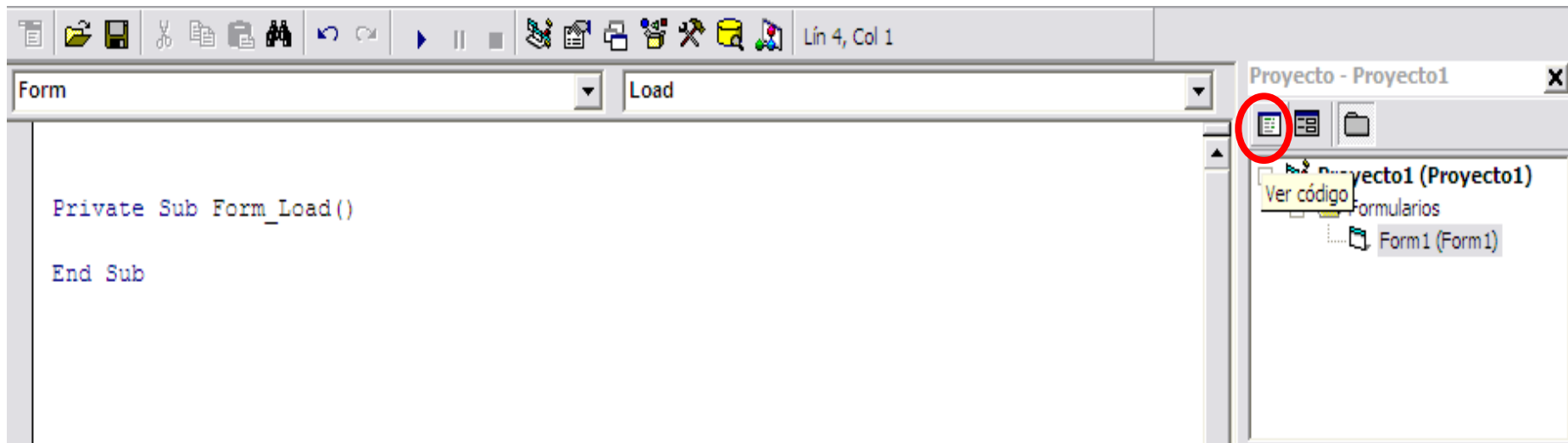
The image shows a Visual Basic IDE workspace. On the left, a form is displayed with a text box at the top and a button labeled "Aceptar" below it. The button is highlighted with a red oval. On the right, the Properties window for the selected button (Command1) is open. The window title is "Propiedades - Command1". The object is identified as "Command1" of type "CommandButton". The "Caption" property is set to "Aceptar" and is also highlighted with a red oval. Other visible properties include "Appearance" (1 - 3D), "BackColor" (&H8000000F), and "Cancel" (False).

Propiedades - Command1	
Command1 CommandButton	
Alfabética Por categorías	
(Nombre)	Command1
Appearance	1 - 3D
BackColor	&H8000000F
Cancel	False
Caption	Aceptar

Visual Basic

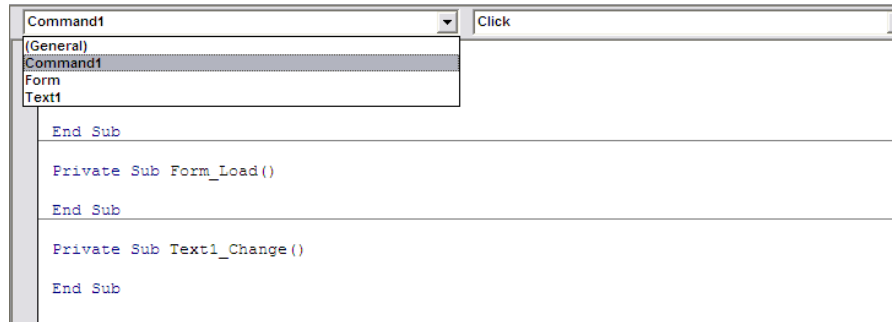
Ejercicio 1

- Para empezar a escribir el código de nuestra aplicación en el **Editor de Código**, podemos tomar 2 caminos:
 - Hacer doble clic sobre el formulario o sobre alguno de los controles con los que estamos trabajando.
 - En la ventana Explorador de Proyectos, seleccionaremos el nombre de un formulario o un módulo y elegiremos **Ver código**.

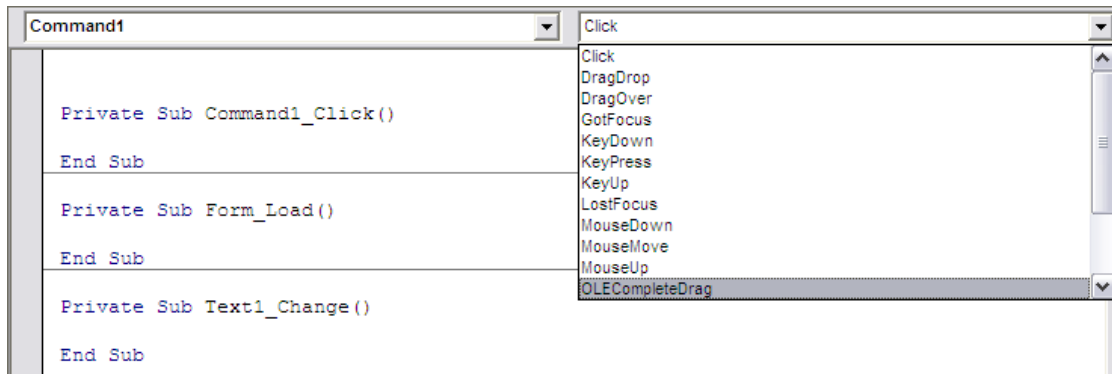


Visual Basic

Ejercicio 1

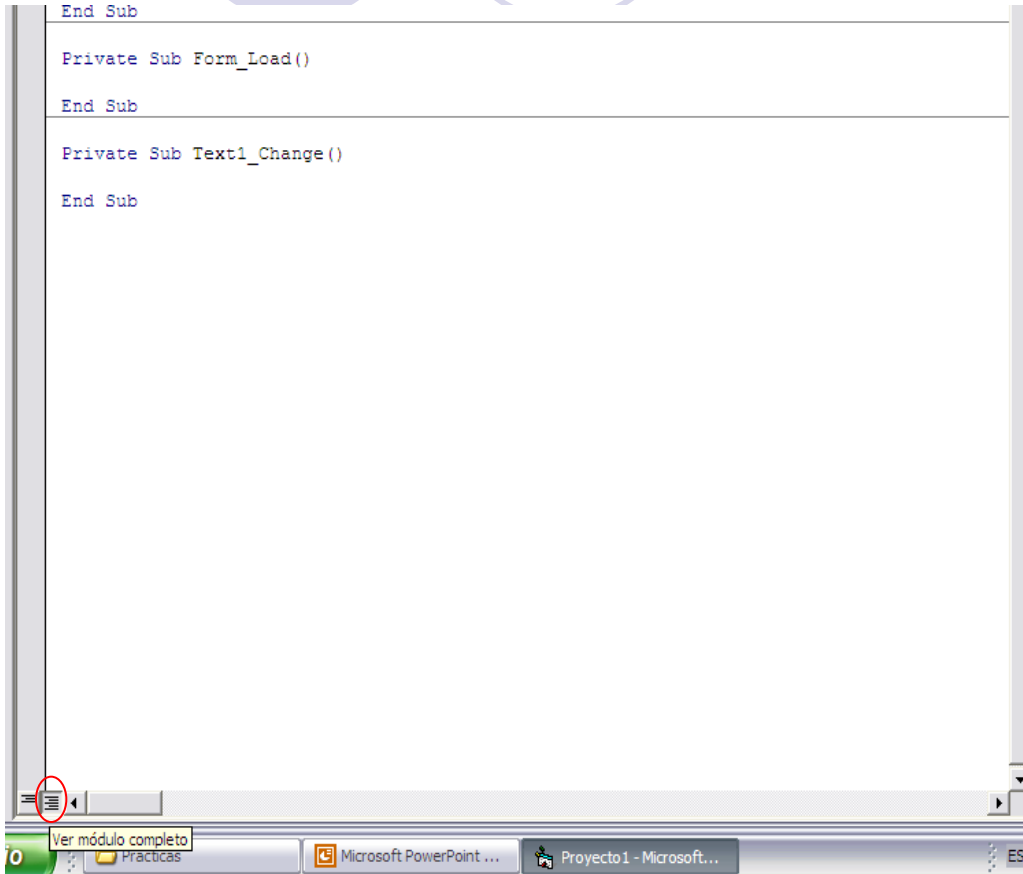


- En la ventana de código podemos elegir el procedimiento sobre el que queremos trabajar.
- También podemos elegir un nuevo evento para trabajar con ese control.



Visual Basic

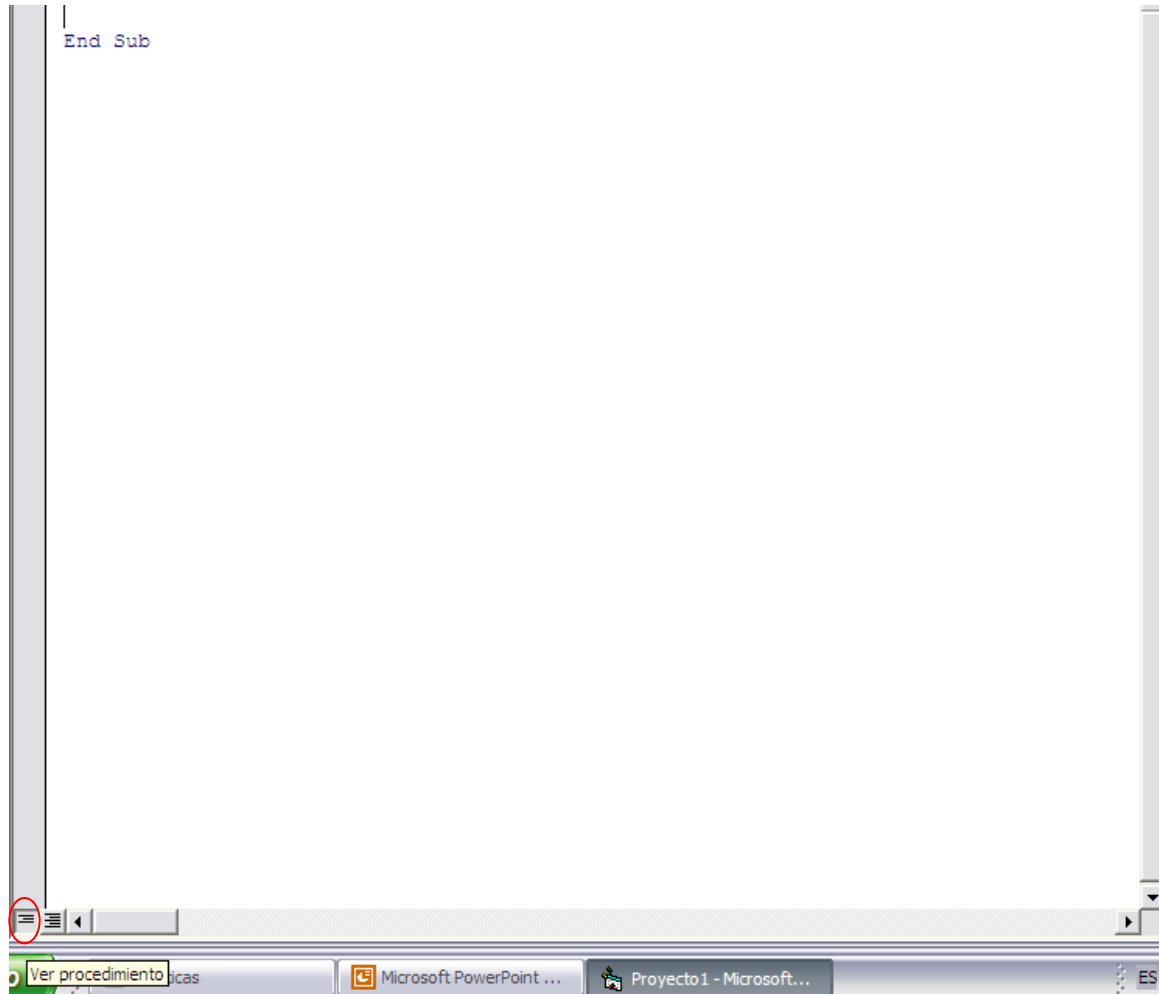
Ejercicio 1



- La presentación del Código en pantalla se puede hacer de varias formas:
 - Podemos ver el Código Completo de la aplicación (la de todos los controles) pulsando la casilla inferior izquierda de “*Ver módulo Completo*”

Visual Basic

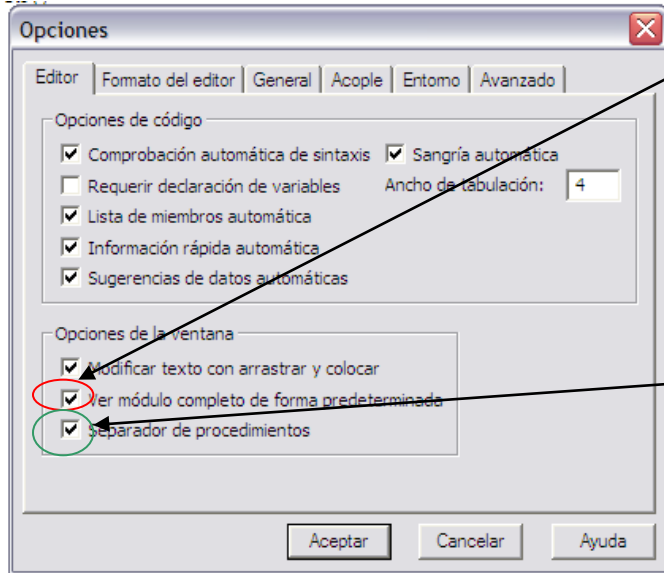
Ejercicio 1



- O ver exclusivamente el código correspondiente al control que tenemos señalado. Esto se hace pulsando la casilla inferior izquierda de “Ver Procedimiento”.

Visual Basic

Ejercicio 1



● Para que siempre podamos presentar todos los procedimientos en la misma ventana de código, podemos activar la opción “*Ver módulo completo de forma predeterminada*”.

● Si queremos quitar la línea que separa cada procedimiento, podemos deshabilitar la opción “*Separador de procedimientos*”.

● *Herramientas* → *Opciones* → *Editor*

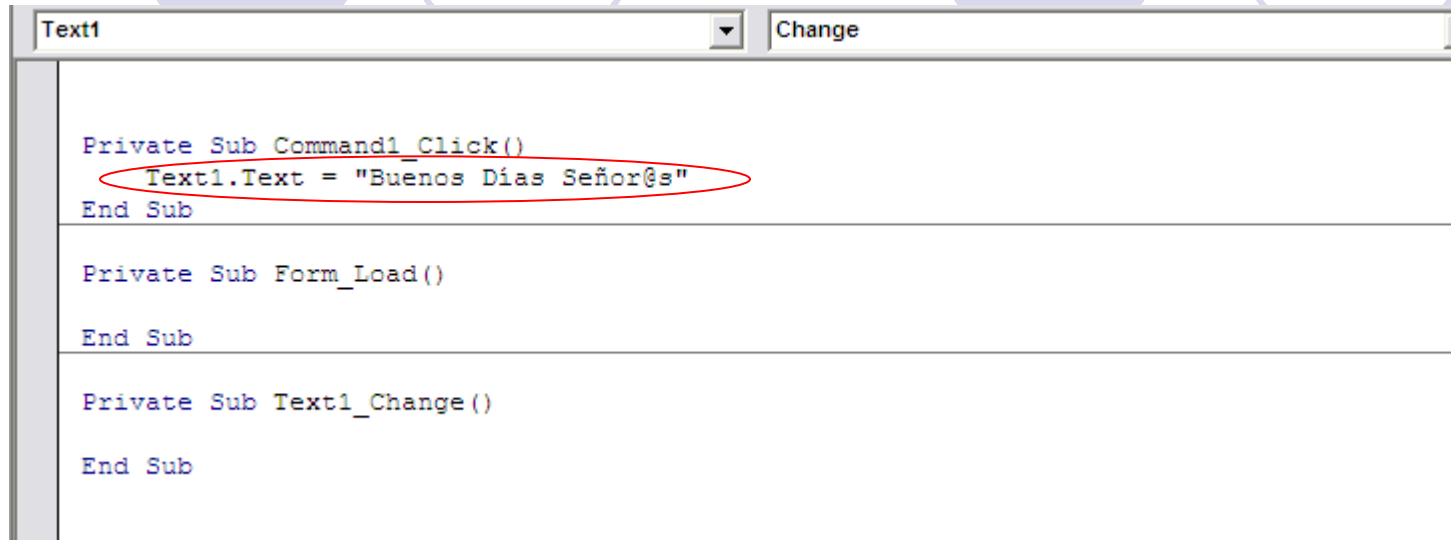
Visual Basic

Ejercicio 1

- El Código en una aplicación de Visual Basic se divide en bloques menores llamados **Procedimientos**. Un ***Procedimiento de Evento***, contiene código que se ejecuta cuando se produce un evento (como cuando un usuario hace clic en un botón). Un *Procedimiento de Evento* de un control combina el **Nombre Real del Control** (especificado en la propiedad **Name**), **un carácter subrayado** y **el Nombre del Evento**. Por ejemplo, si queremos que un botón de comando llamado *Command1* invoque un *Procedimiento de Evento* cuando se haga **clic** en él, utilizaremos el procedimiento: **Command1_Click**.

Visual Basic

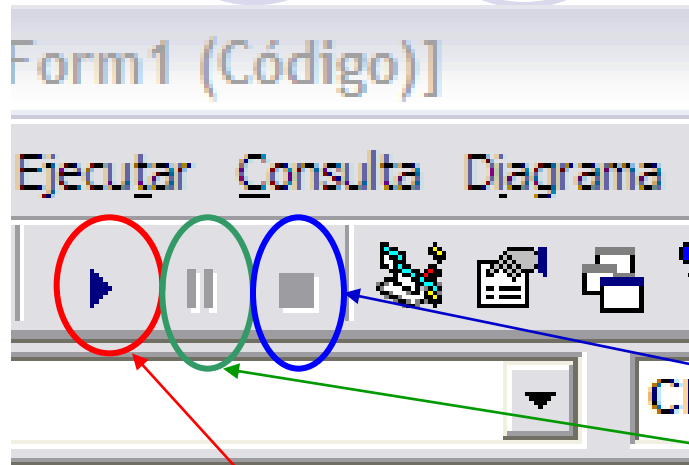
Ejercicio 1



- Escribimos el código entre el *Sub* y *End Sub* del procedimiento *Command1*. Escribimos: ***Text1.Text = "Buenos Días Señor@s"***.
- Hay que observar que aquí el código simplemente cambia la propiedad **Text** del control llamado *Text1* para que sea *Buenos Días Señor@s*. La sintaxis de este ejemplo tiene el formato: **objeto.propiedad**, donde *Text1* es el **objeto** y *Text* es la **propiedad**. Podemos utilizar esta sintaxis para cambiar los valores de las propiedades de cualquier formulario o control como respuesta a los eventos que se producen mientras se ejecuta nuestra aplicación.

Visual Basic

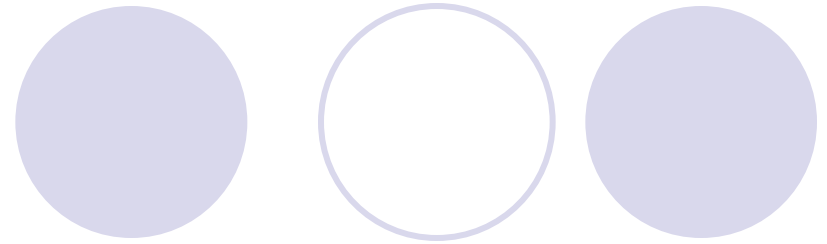
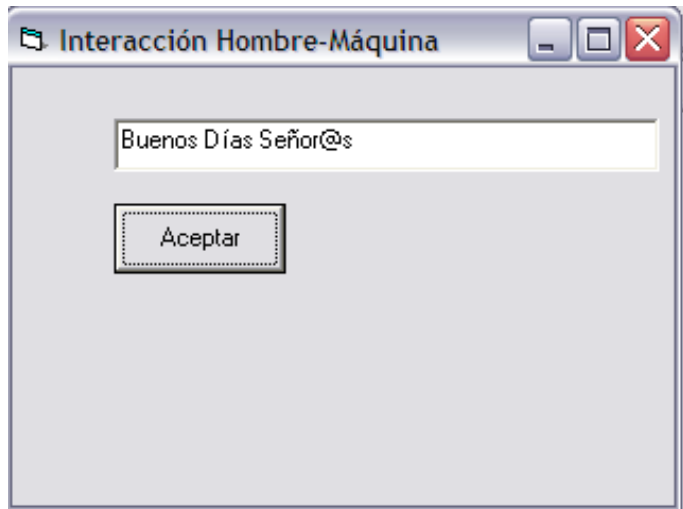
Ejercicio 1



- Para **iniciar la ejecución de nuestra aplicación**, **páusar** o **parar su ejecución**, utilizaremos estos botones.

Visual Basic

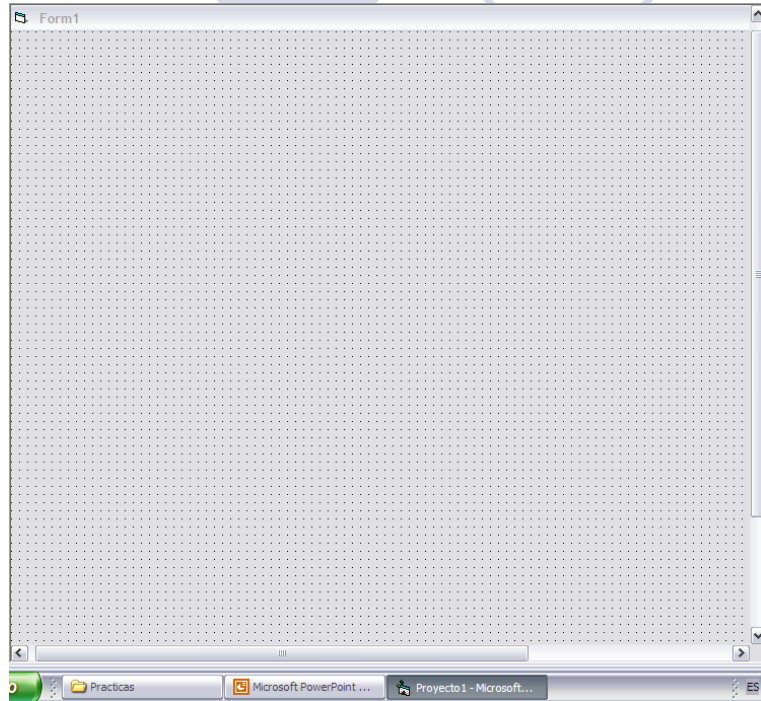
Ejercicio 1



- Principio de la ejecución.
- Final de la ejecución después de pulsa la tecla *Aceptar*.

Visual Basic

El Formulario



- Es el punto a partir del cual se desarrolla cualquier aplicación Visual Basic. Básicamente se trata de una ventana estándar Windows, en la que nosotros podemos insertar los controles que necesitemos para presentar datos, solicitarlos o definir ciertos procedimientos.

Visual Basic

El Formulario

- Por defecto, el título del formulario es **Form1**, al igual que su nombre. Estos dos términos, título y nombre, son fáciles de confundir. La mayoría de controles tienen un título, que podemos obtener y modificar mediante la propiedad **Caption**, que no es más que el texto que aparece en el control, y no tiene en realidad ninguna correspondencia ni referencia con el código de la aplicación.
- El nombre del control o formulario, por el contrario, sí que es referenciado en el código cada vez que se quiere manipular el objeto. El nombre de un control, que se establece durante el diseño de la aplicación, no será modificado posteriormente, y se utilizará tanto para acceder a las propiedades de ese objeto como para realizar cualquier otro tipo de operaciones con él. La propiedad correspondiente al nombre del objeto es **Name**.

Visual BasicEl Formulario

Propiedad	Definición
<i>Auto3D</i>	Por defecto esta propiedad toma el valor <i>False</i> , por que no se dará ningún efecto tridimensional al formulario. Si le damos el valor <i>True</i> , el fondo de la ventana se pondrá gris, y los controles tomarán un nuevo estilo apareciendo como esculpidos en la ventana. Si además de dar el valor <i>True</i> a esta propiedad seleccionamos un tipo de borde grueso no redimensionable, con la propiedad <i>BorderStyle</i> , también el título aparecerá con un efecto tridimensional.
<i>AutoRedraw</i>	Dando el valor <i>True</i> a esta propiedad, que por defecto toma <i>False</i> , conseguiremos que todas las acciones de impresión y dibujo sobre el formulario no vayan directamente a pantalla, sino a un mapa de puntos mantenido en memoria, a partir del cual se dibujará en pantalla. Esta técnica permite recuperar el contenido completo del formulario, aun cuando éste se haya perdido en pantalla y no exista ningún control asociado, porque en realidad el aspecto que debe tener el formulario no se genera directamente sobre ésta, sino que se encuentra almacenado en memoria.
<i>BackColor</i>	Establece el color de fondo del formulario.
<i>BorderStyle</i>	Modifica el tipo de borde de la ventana. Por defecto toma el valor <i>Sizable</i> , por lo que el formulario cuenta con un borde grueso que además permite que la ventana sea cambiada de tamaño. Valores: <i>None</i> : La Ventana no tiene borde alrededor. <i>Fixed single</i> : Borde simple no redimensionable. <i>Sizable</i> : Borde grueso redimensionable. <i>Fixed double</i> : Borde grueso no redimensionable.
<i>Caption</i>	Título que podemos obtener y modificar por medio de esta propiedad. Es el texto que aparece en el control, y no tiene en realidad ninguna correspondencia ni referencia en el código de la aplicación.
<i>ControlBox</i>	Esta propiedad, que por defecto tiene el valor <i>True</i> , es la responsable de que la ventana cuente con un menú de sistema. Simplemente estableciendo esta propiedad a <i>False</i> , eliminaremos dicho menú.

Visual Basic

El Formulario

Propiedad	Definición
<i>Enabled</i>	El valor por defecto de esta propiedad es <i>True</i> , indicando que en principio el formulario está activado, y por lo tanto el usuario puede interactuar con él. Dándole el valor <i>False</i> desactivaremos el formulario, impidiendo así que el usuario trabaje con él.
<i>ForeColor</i>	Establece el color del primer plano en el formulario.
<i>Height</i>	Contiene la altura del formulario. Habitualmente esta propiedad nunca se rellena a mano, sino que toma su valor automáticamente cuando redimensionamos el formulario.
<i>Icon</i>	Podemos asignar a esta propiedad el nombre de un archivo conteniendo un icono, o bien el icono directamente, que será visualizado cuando el formulario de nuestra aplicación se minimice.
<i>KeyPreview</i>	Cuando en un formulario que contiene varios controles se pulsa una tecla, normalmente es el control que está activo el que recibe esa pulsación, y en caso de no ser procesada se pasa al formulario. Dando a esta propiedad el valor <i>True</i> , por defecto toma <i>False</i> , indicaremos que las pulsaciones se pasen primero al formulario, en lugar de al control que está activo.
<i>Left</i>	Esta propiedad contiene la posición horizontal del formulario. Se rellena automáticamente cuando desplazamos el formulario, aunque también es posible asignarle directamente un valor.
<i>MaxButton</i>	Activa y desactiva el botón de maximizar. Por defecto viene a <i>True</i> .
<i>MinButton</i>	Activa y desactiva el botón de minimizar. Por defecto viene a <i>True</i> .
<i>MousePointer</i>	Con esta propiedad podemos fijar el tipo de icono para el puntero del ratón cuando éste se encuentre sobre nuestra ventana. Podemos elegir entre una serie de iconos predefinidos, o bien elegir la opción <i>Icon</i> para a continuación indicar con la propiedad <i>MouseIcon</i> el nombre del archivo que contiene el icono que se utilizará.

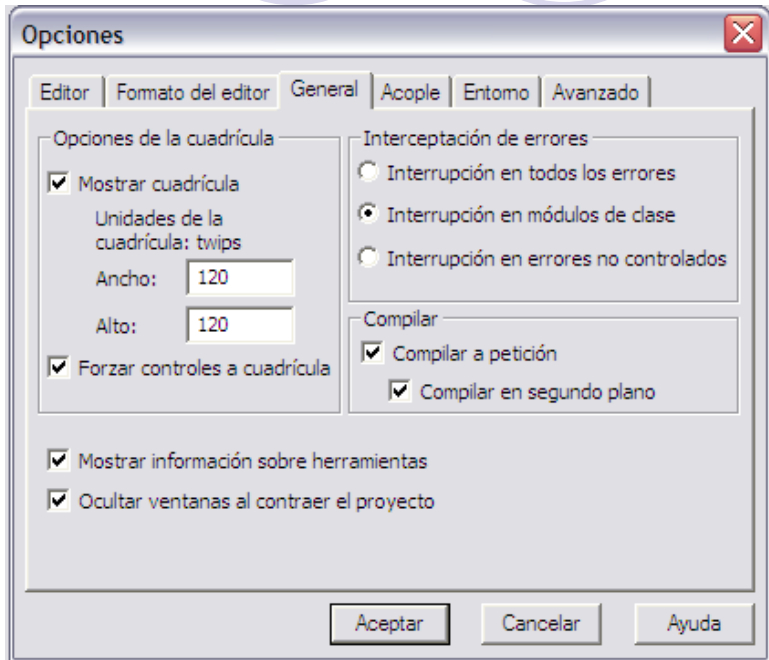
Visual Basic

El Formulario

Propiedad	Definición
<i>Name</i>	Establece el nombre del control o formulario que es referenciado en el código cada vez que se quiere manipular ese objeto. El nombre de un control, que se establece durante el diseño de una aplicación, no será modificado posteriormente, y se utilizará tanto para acceder a las propiedades de ese objeto como para realizar cualquier otro tipo de operación con él.
<i>Picture</i>	Por medio de esta propiedad podemos establecer un gráfico como fondo de formulario.
<i>Tag</i>	Nos permite asociar al formulario cualquier información que necesitemos.
<i>Top</i>	Establece la posición vertical del formulario, y junto con <i>Left</i> conforma la coordenada de la esquina superior izquierda de éste.
<i>Visible</i>	Normalmente esta propiedad siempre toma el valor <i>True</i> en diseño de tamaño, indicando así que el formulario estará visible cuando se ejecute la aplicación. Asignándole el valor <i>False</i> en cualquier momento, conseguiremos que el formulario no sea visible.
<i>Width</i>	Almacena la anchura del formulario, y junto con <i>Height</i> fija las dimensiones del formulario.
<i>WindowState</i>	Cuando se ejecuta una aplicación, habitualmente el formulario aparece en pantalla en la posición y con las dimensiones que se establecieron durante el diseño. Este es el estado <i>Normal</i> de la ventana, que equivale al valor 0 de esta propiedad. Asignándole el valor 1, la ventana aparecerá inicialmente minimizada (<i>Minimized</i>), como un icono, y usando el valor 2 conseguiremos que la ventana esté inicialmente maximizada (<i>Maximized</i>), ocupando toda la pantalla.

Visual Basic

El Formulario



- Herramientas → Opciones → General
- Esta opción nos permite elegir entre que la rejilla del formulario, en tiempo de diseño, aparezca o no, marcando o no la opción “*Mostrar cuadrícula*”, así como fijar el número de “*twips*”, tanto en sentido horizontal (*Ancho*) como vertical (*Alto*), que habrá de distancia entre un punto y otro. Por último podemos, con la opción “*Forzar controles a cuadrícula*”, indicar a Visual Basic que cada vez que inserte o mueva un control, lo ajuste lo más posible a la matriz de puntos.

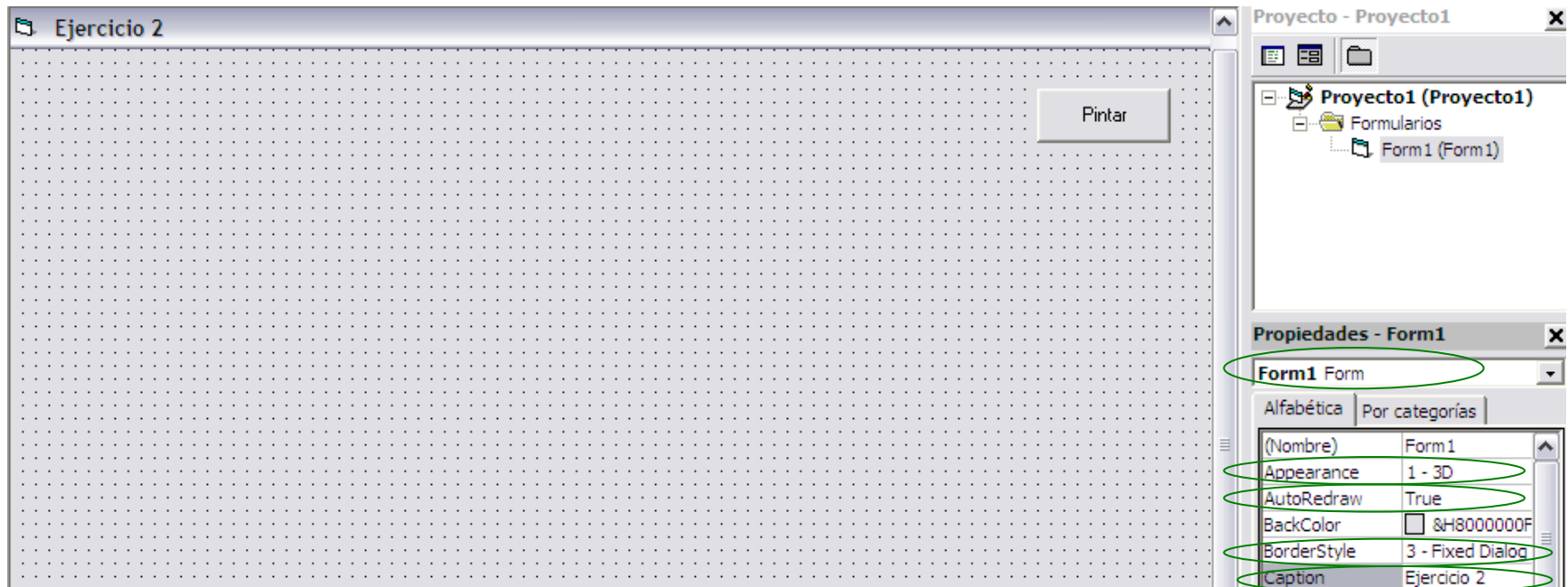
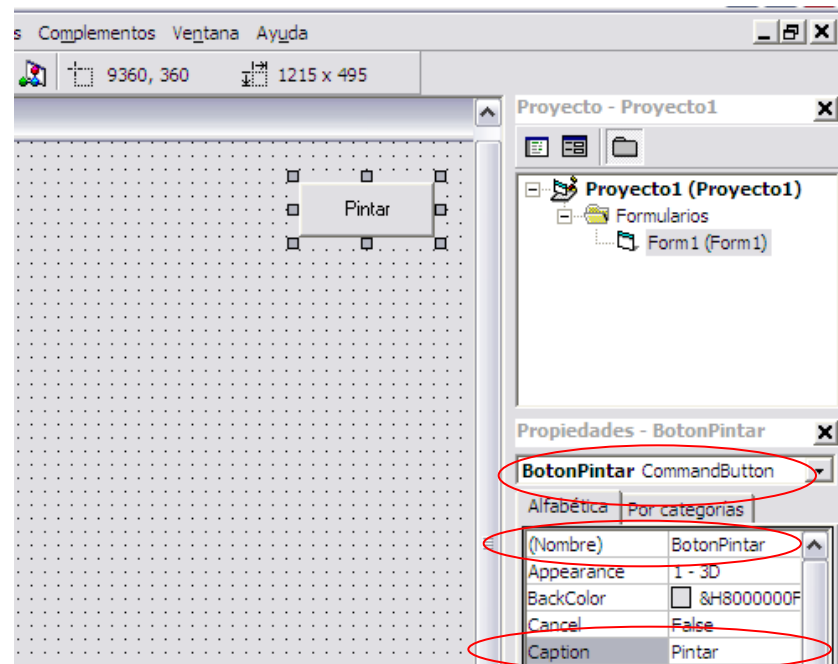
Visual Basic

Ejercicio 2

- Consiste en un botón de comando (**CommandButton**) que cada vez que lo pulsemos visualizará en el formulario: “Estamos Pintando”
- Vamos a fijar las propiedades del Formulario:
 - *Appearance*: **1 – 3D**.
 - *AutoRedraw*: **True**.
 - *BorderStyle*: **Fixed Dialog**.
 - *Caption*: **Ejercicio 2**.
- Vamos a fijar las propiedades del Botón de Comando:
 - *Caption*: **Pintar**.
 - *Name*: **BotonPintar**.

Visual Basic

Ejercicio 2



Ejercicio 2

```
Private Sub BotonPintar_Click()
```

```
    Print "Estamos Pintando"
```

'Cada vez que pulsemos la tecla Pintar
' se visualizará en el formulario:
' Estamos Pintando.

```
End Sub
```

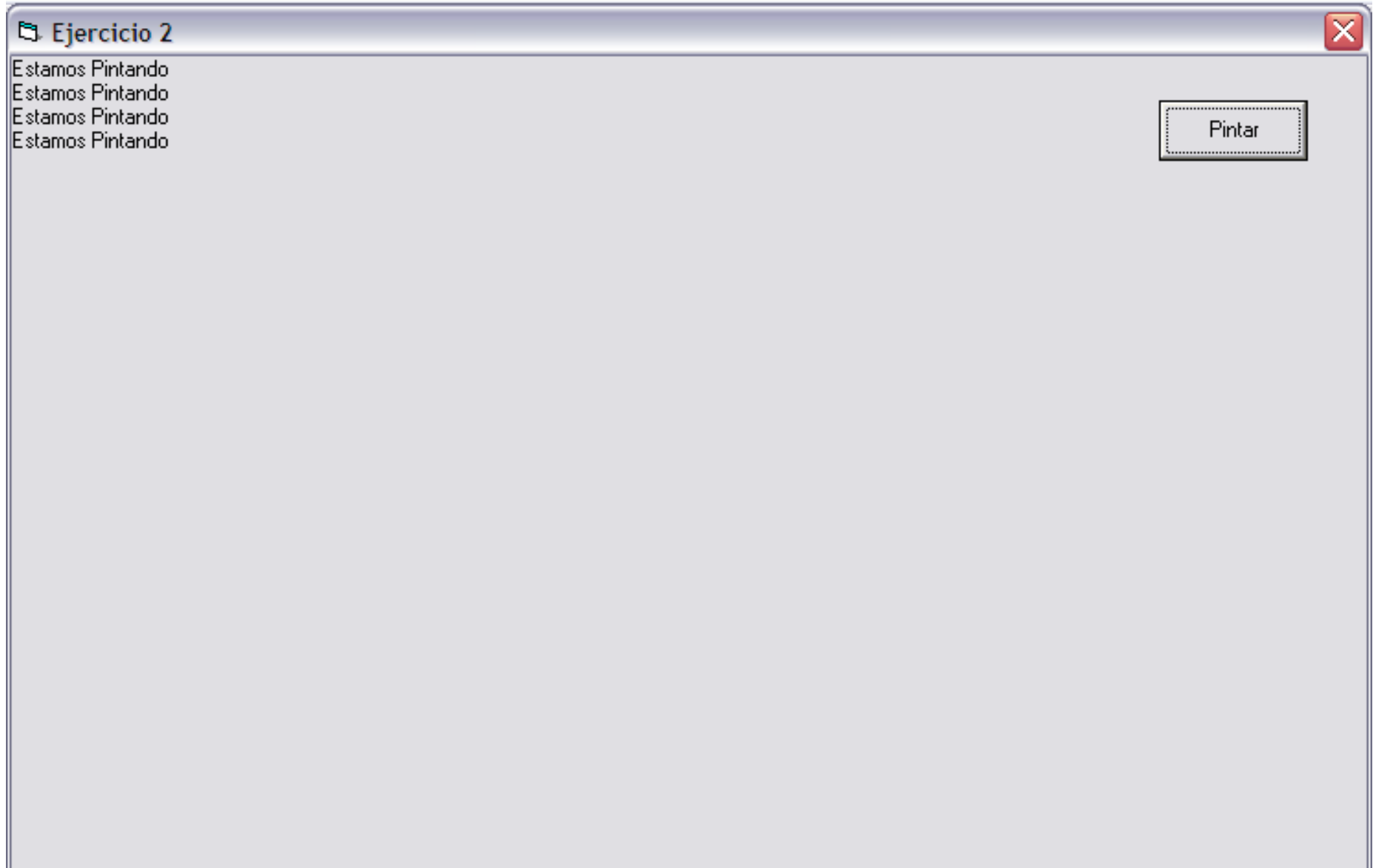
```
Private Sub Form_Load()
```

```
End Sub
```

Comentarios

Visual Basic

Ejercicio 2



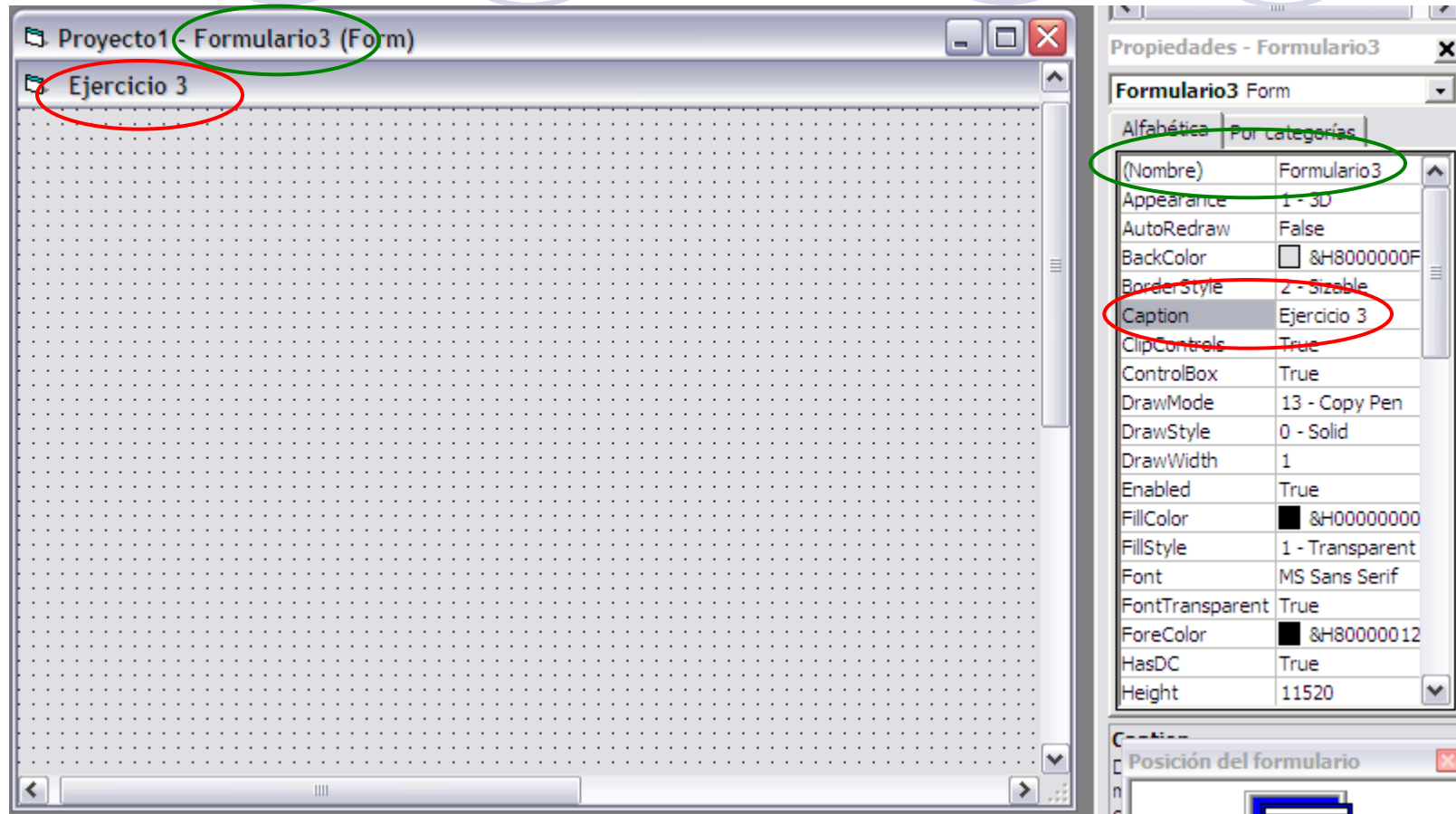
Visual BasicEjercicio 3

- Consiste en una aplicación que, cuando pulsemos el botón “Pulsa aquí”, en la caja de texto tiene que aparecer el mensaje “Estamos de Prácticas”; cuando pulsemos el botón “?”, en la caja de texto tiene que aparecer el mensaje “¿Contentos?; y cuando pulsemos el botón “Salir”, la aplicación finalizará.

Objeto	Propiedad	Valor
Formulario	Captión	Ejercicio 3
	Nombre	Formulario3
Cuadro de Texto	BackColor	&H00C0FFFF&
	Text	(En blanco)
	Nombre	Mensaje
Botón “Pulsa aquí”	Caption	Pulsa aquí
	Nombre	BotonPulsar
Botón “?”	Caption	?
	Nombre	BotonX
Botón “Salir”	Caption	Salir
	Nombre	BotonSalir

Visual Basic

Ejercicio 3



Visual Basic

Ejercicio 3

The image shows a screenshot of the Visual Basic IDE. The main window is titled "Proyecto1 - Formulario3 (Form)" and contains a form with a single text box labeled "Text1". The text box is highlighted in yellow. To the right of the form is the "Propiedades - Mensaje" (Properties - Message) window, which displays the properties of the selected text box. The properties are listed in a table with two columns: "Nombre" (Name) and "Mensaje" (Message). The properties are: (Nombre) Mensaje, Alignment 0 - Left Justify, Appearance 1 - 3D, BackColor &H00C0FFFF, BorderStyle 1 - Fixed Single, CausesValidation True, DataField, DataFormat, DataMember, DataSource, DragIcon (Ninguno), DragMode 0 - Manual, Enabled True, Font MS Sans Serif, ForeColor &H80000008, Height 495, HelpContextID 0, HideSelection True, Index, TabStop True, Tag, Text, ToolTipText, and Top 480. The properties "BackColor", "BorderStyle", and "Text" are circled in red. Below the properties window is the "Data Sources" window, which shows the "Posición del formulario" (Form Position) window.

Nombre	Mensaje
(Nombre)	Mensaje
Alignment	0 - Left Justify
Appearance	1 - 3D
BackColor	&H00C0FFFF
BorderStyle	1 - Fixed Single
CausesValidation	True
DataField	
DataFormat	
DataMember	
DataSource	
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif
ForeColor	&H80000008
Height	495
HelpContextID	0
HideSelection	True
Index	
TabStop	True
Tag	
Text	
ToolTipText	
Top	480

Visual Basic

Ejercicio 3

The image shows the Visual Basic IDE with a form titled "Proyecto1 - Formulario3 (Form)". Inside the form, there is a yellow rectangular box at the top and a button labeled "Pulsa aquí" below it. The button is highlighted with a red oval. To the right of the form is the "Propiedades - BotonPulsar" window, which displays the properties of the selected button. The "Caption" property is highlighted with a red oval and set to "Pulsa aquí". The "Name" property is highlighted with a green oval and set to "BotonPulsar". The "Appearance" property is set to "1 - 3D". The "CausesValidation" property is set to "True".

Proyecto1 - Formulario3 (Form)

Ejercicio 3

Pulsa aquí

Propiedades - BotonPulsar

BotonPulsar CommandButton

Alfabética | Por Categorías

(Nombre)	BotonPulsar
Appearance	1 - 3D
BackColor	&H8000000F
Cancel	False
Caption	Pulsa aquí
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif
Height	495
HelpContextID	0
Index	
Left	3480
MaskColor	&H00C0C0C0
MouseIcon	(Ninguno)

Posición del formulario

Visual Basic

Ejercicio 3

Proyecto1 - Formulario3 (Form)

Ejercicio 3

Pulsa aquí

?

Propiedades - BotonX

BotonX CommandButton

Alfabética Por Categorías

(Nombre)	BotonX
Appearance	1 - 3D
BackColor	<input checked="" type="checkbox"/> &H8000000F
Cancel	False
Caption	?
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno) ...
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif
Height	495
HelpContextID	0
Index	
Left	3480
MaskColor	<input checked="" type="checkbox"/> &H00C0C0C0
MouseIcon	(Ninguno)

Posición del formulario

Visual Basic

Ejercicio 3

The image shows a screenshot of the Visual Basic IDE. The main window is titled "Proyecto1 - Formulario3 (Form)". Inside the form, there is a yellow rectangular area at the top. Below it, there is a button labeled "Pulsa aquí". Further down, there is a button labeled "Salir" which is circled in red. To the left of the "Salir" button is a small button with a question mark. The Properties window on the right is titled "Propiedades - BotonSalir". It shows the properties for the "BotonSalir" CommandButton. The "Caption" property is set to "Salir" and is circled in red. The "Name" property is set to "BotonSalir" and is circled in green. The "CausesValidation" property is set to "True".

Proyecto1 - Formulario3 (Form)

Ejercicio 3

Pulsa aquí

?

Salir

Propiedades - BotonSalir

BotonSalir CommandButton


Alfabética Por Categorías

Propiedad	Valor
(Nombre)	BotonSalir
Appearance	1 - 3D
BackColor	&H8000000F
Cancel	False
Caption	Salir
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif
Height	495
HelpContextID	0
Index	
Left	4440
MaskColor	&H00C0C0C0
MouseIcon	(Ninguno)

Posición del formulario

Visual Basic

Ejercicio 3



```
Private Sub BotonPulsar_Click()  
    Mensaje.Text = "Estamos en Prácticas" 'Cuando pulsemos este botón, saldrá  
                                           ' este mensaje "Estamos en Prácticas"  
End Sub
```

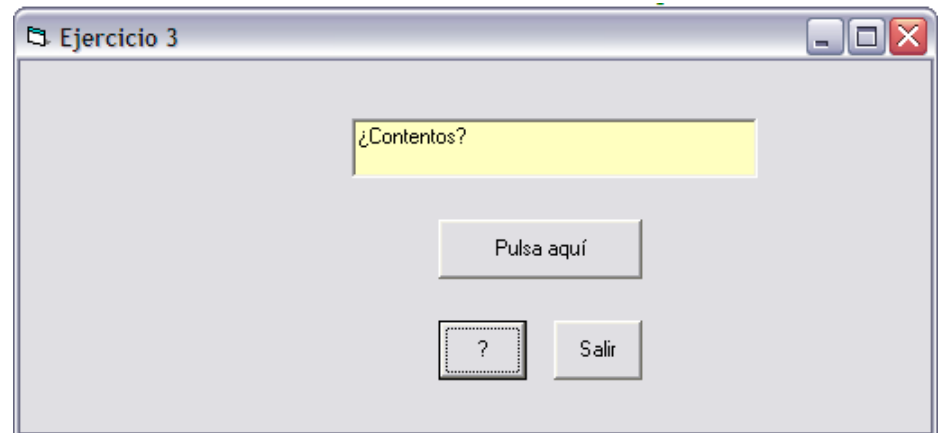
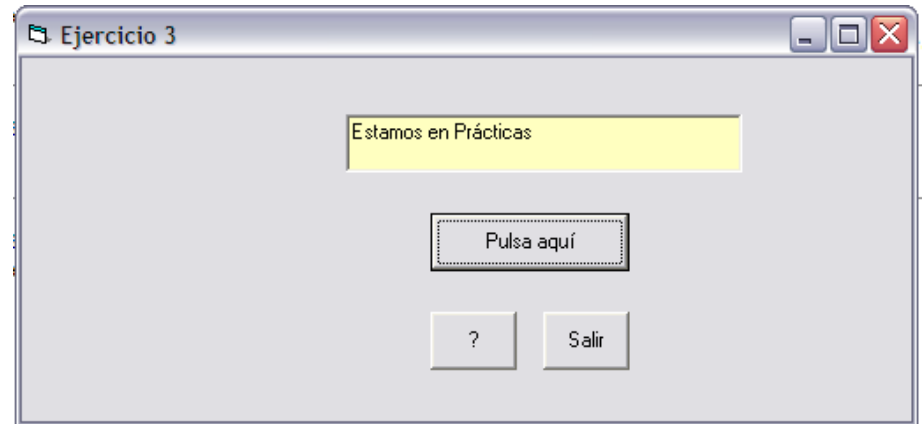
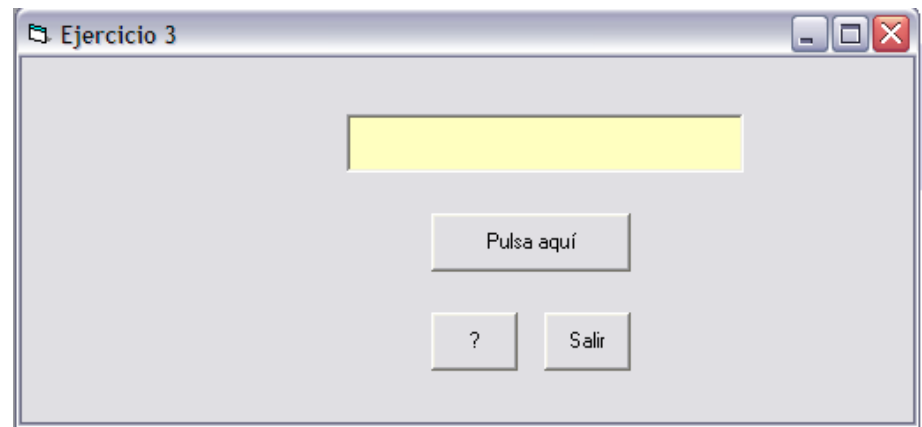
```
Private Sub BotonSalir_Click()  
    End 'Finaliza la aplicación  
End Sub
```

```
Private Sub BotonX_Click()  
    Mensaje.Text = "¿Contentos?" 'Cuando pulsemos este botón, saldra"  
                                'este mensaje "¿Contentos?"  
End Sub
```

,

Visual Basic

Ejercicio 3



Visual Basic

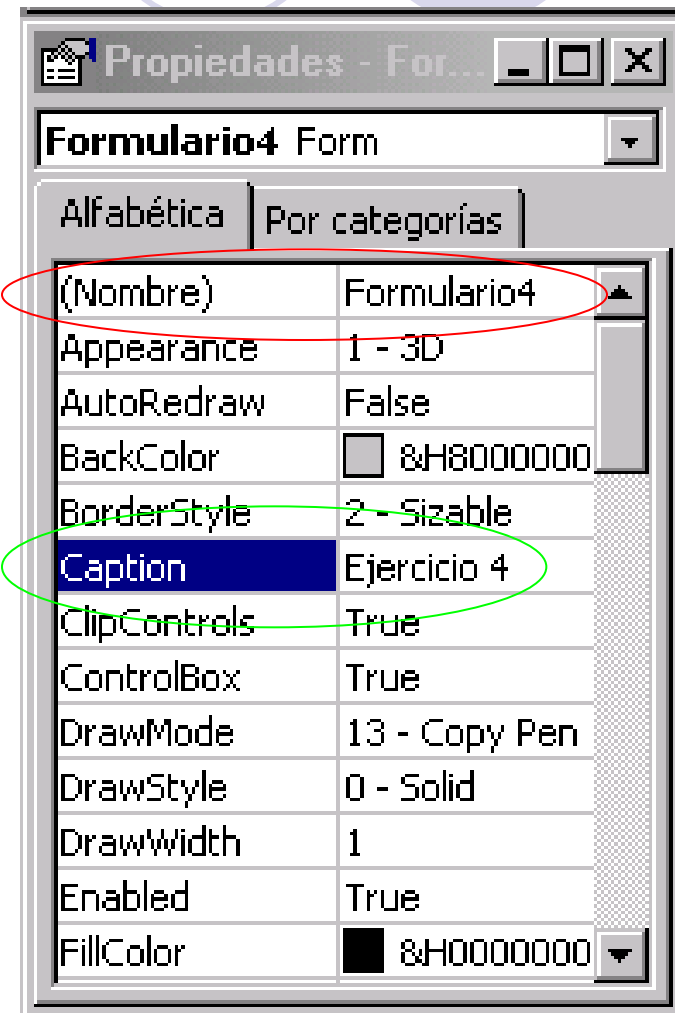
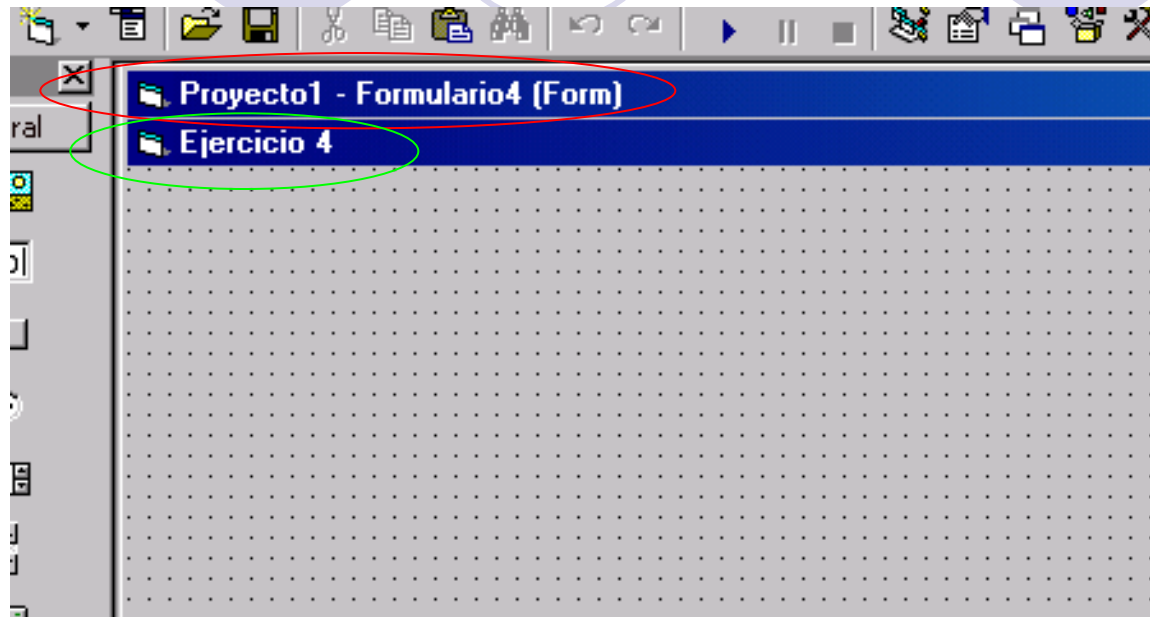
Ejercicio 4

- Consiste en una aplicación que, cuando pulsemos el botón “Rojo”, la caja de texto se pone roja, cuando pulsemos el botón “Azul”, la caja de texto se pone azul y cuando pulsemos el botón “Verde” la caja de texto se pone verde. Pulsaremos “Salir” para salir de la aplicación.

Objeto	Propiedad	Valor
Formulario	Caption	Ejercicio 4
	Nombre	Formulario 4
Cuadro de Texto	Text	(En blanco)
	Nombre	Caja
Botón “Rojo”	Caption	Rojo
	Nombre	Boton_Rojo
Botón “Verde”	Caption	Verde
	Nombre	Boton_Verde
Botón “Azul”	Caption	Azul
	Nombre	Boton_Azul
Botón “Salir”	Caption	Salir
	Nombre	Boton_Salir

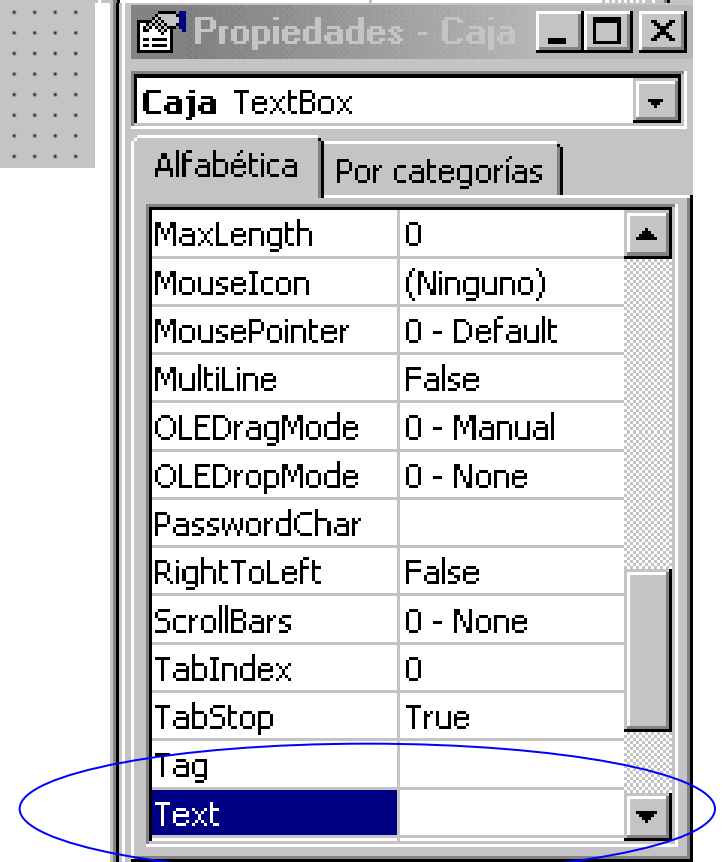
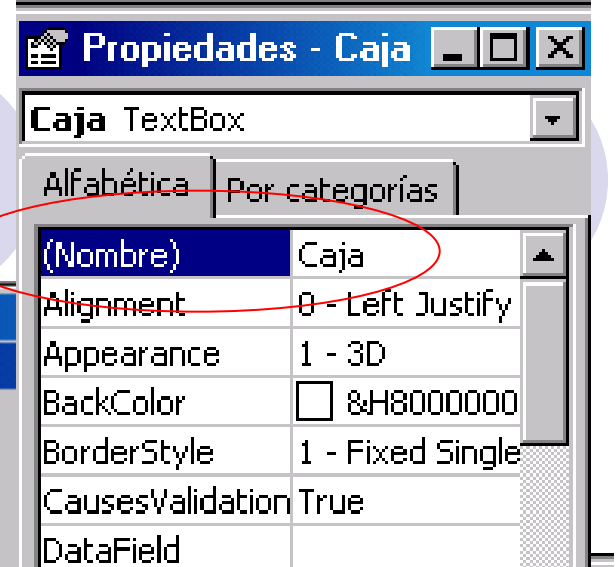
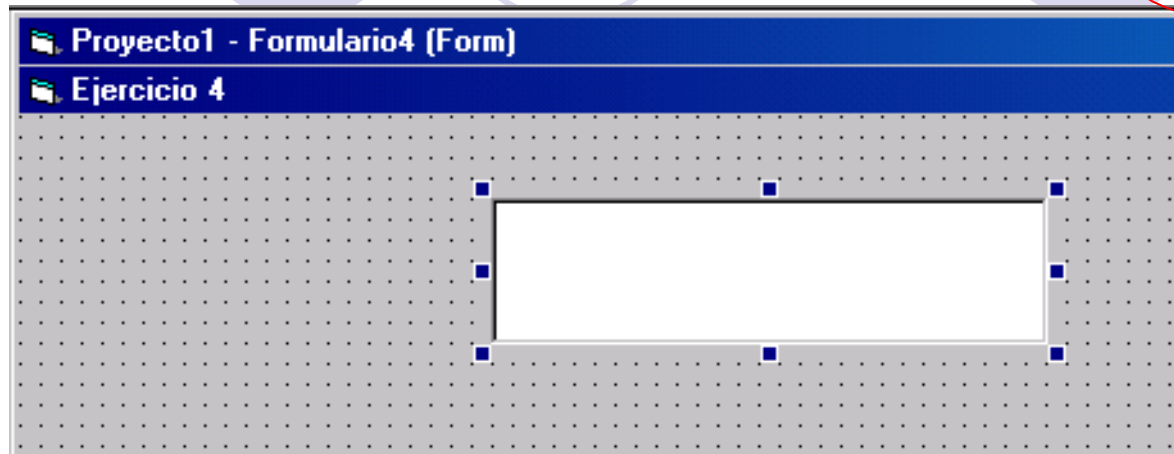
Visual Basic

Ejercicio 4



Visual Basic

Ejercicio 4

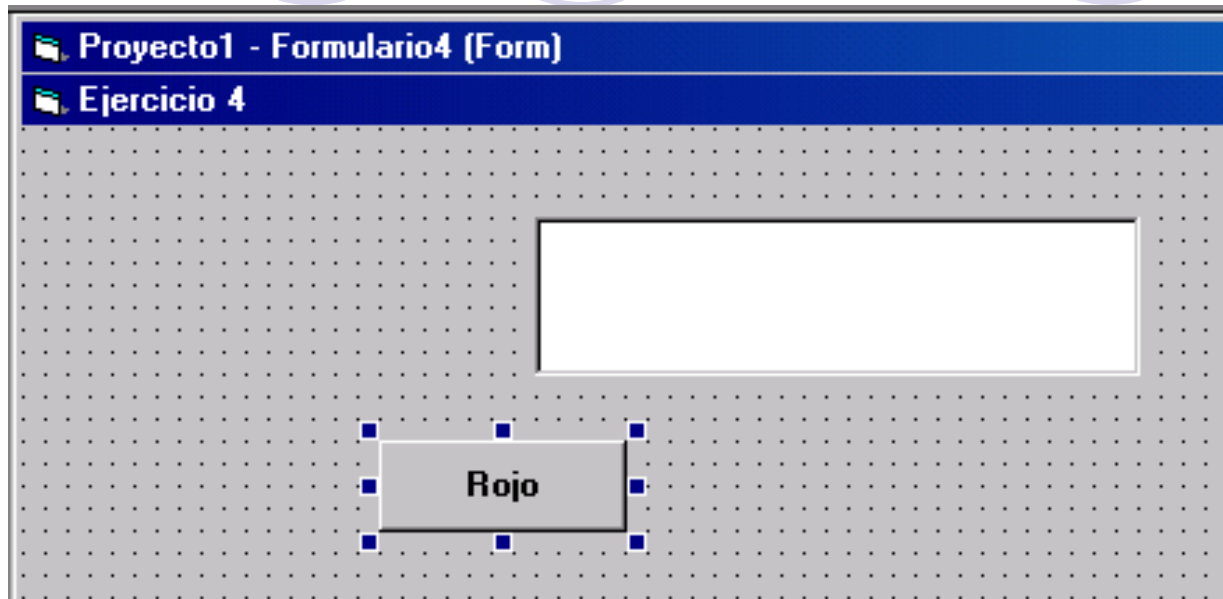


Visual Basic

Ejercicio 4

Proyecto1 - Formulario4 (Form)


Ejercicio 4



Propiedades - Bo...

Boton_Rojo CommandButton

Alfabética | Por categorías

(Nombre)	Boton_Rojo
Appearance	1 - 3D
BackColor	 &H00C0C0C
Cancel	False
Caption	Rojo
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif

Caption

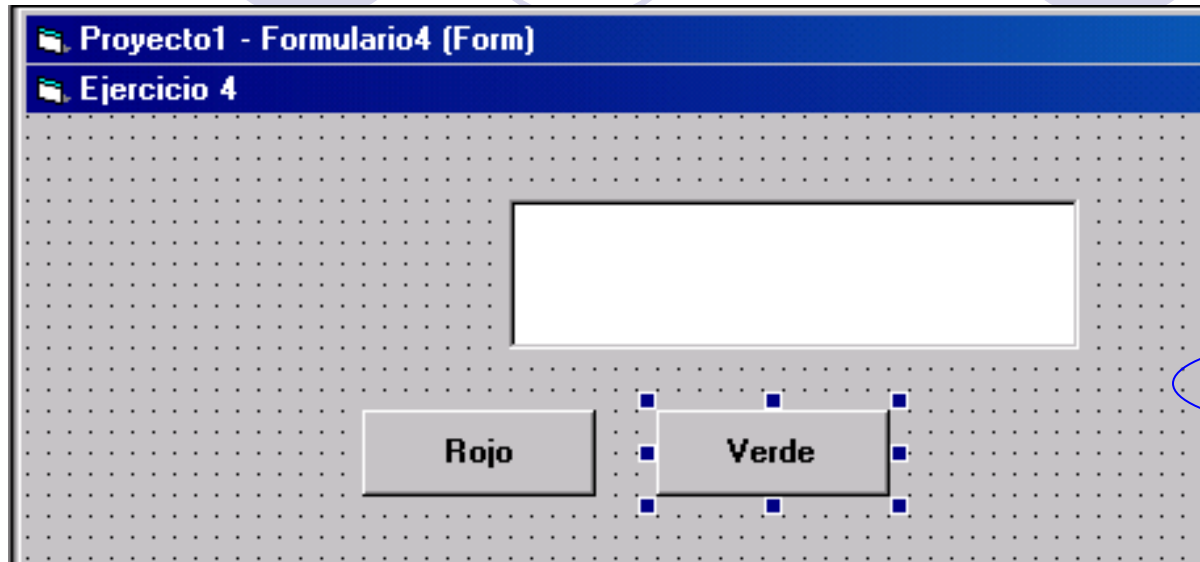
Devuelve o establece el texto mostrado en la barra de título de un

Visual Basic

Ejercicio 4


Proyecto1 - Formulario4 (Form)

Ejercicio 4



Propiedades - Boton_Verde CommandButton

Alfabética | Por categorías

(Nombre)	Boton_Verde
Appearance	1 - 3D
BackColor	 &H80000000
Cancel	False
Caption	Verde
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif

Visual Basic

Ejercicio 4

Proyecto1 - Formulario4 (Form)

Ejercicio 4

Rojo Verde Azul

Propiedades - Bo...

Boton_Azul CommandButton

Alfabética Por categorías

(Nombre)	Boton_Azul
Appearance	1 - 3D
BackColor	&H80000000
Cancel	False
Caption	Azul
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif

Visual Basic

Ejercicio 4

Proyecto1 - Formulario4 (Form)

Ejercicio 4

[Empty Text Box]

Rojo Verde Azul

Salir

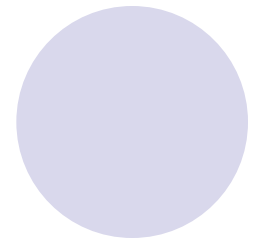
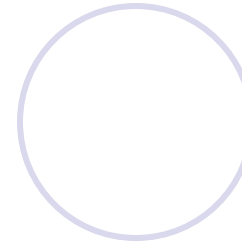
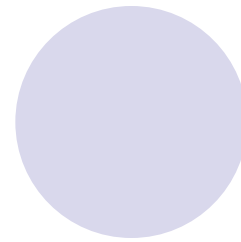
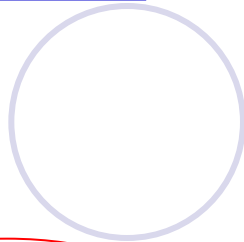
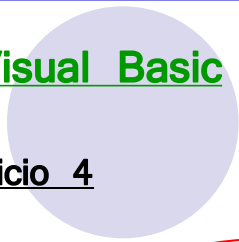
Propiedades - Boton_Salir CommandButton

Alfabética Por categorías

(Nombre)	Boton_Salir
Appearance	1 - 3D
BackColor	&H80000000
Cancel	False
Caption	Salir
CausesValidation	True
Default	False
DisabledPicture	(Ninguno)
DownPicture	(Ninguno)
DragIcon	(Ninguno)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif

Visual Basic

Ejercicio 4



```
Private Sub Boton_Azul_Click()  
    Caja.BackColor = vbBlue  
  
End Sub
```

'Cuando pulsemos este botón, la
'caja de texto "Caja" se pondrá Azul.

```
Private Sub Boton_Rojo_Click()  
    Caja.BackColor = vbRed  
  
End Sub
```

'Cuando pulsemos este botón, la
'caja de texto "Caja" se pondrá Roja.

```
Private Sub Boton_Salir_Click()  
    End  
  
End Sub
```

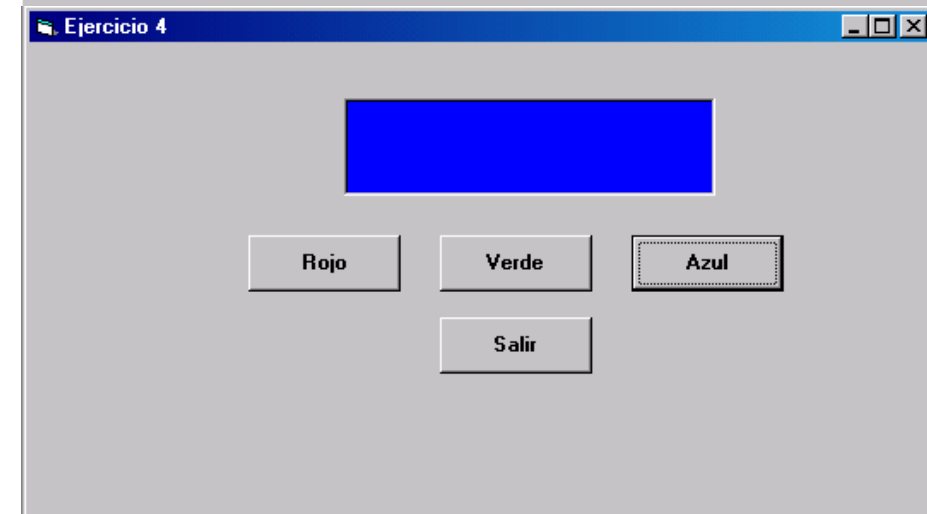
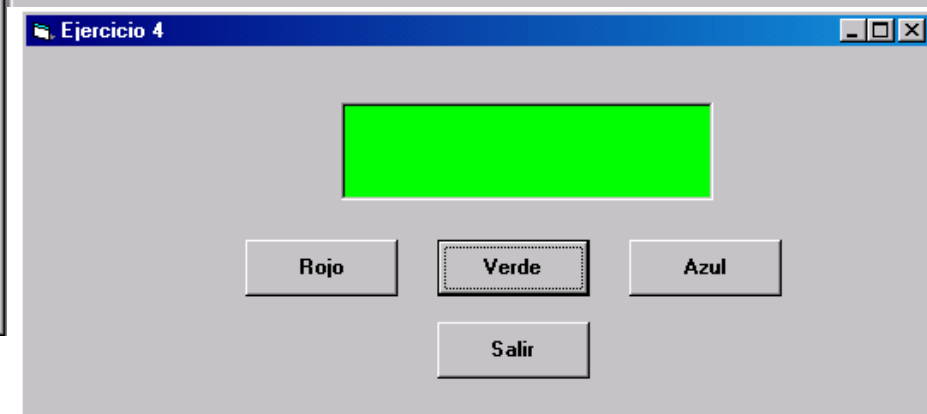
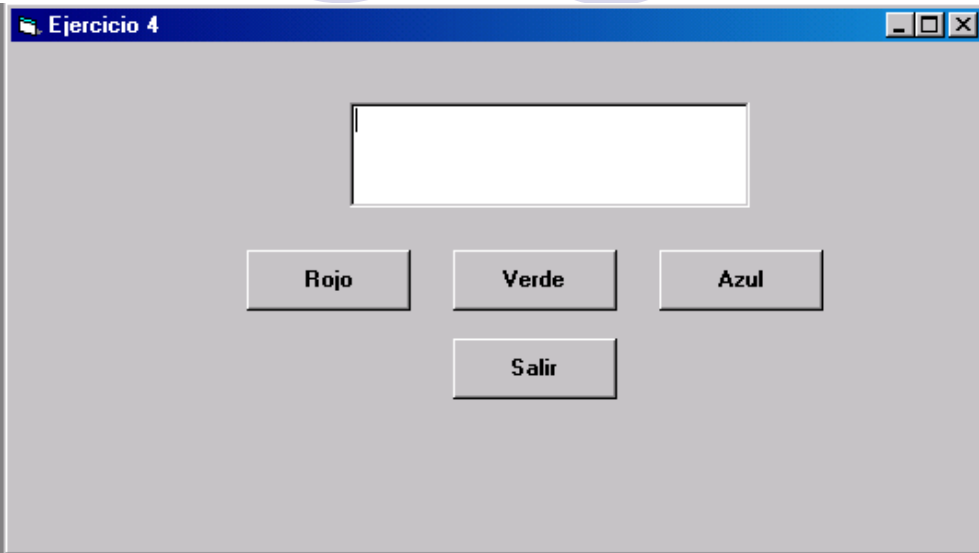
'Finaliza la aplicación.

```
Private Sub Boton_Verde_Click()  
    Caja.BackColor = vbGreen  
  
End Sub
```

'Cuando pulsemos este botón, la
'caja de texto "Caja" se pondrá Verde.

Visual Basic

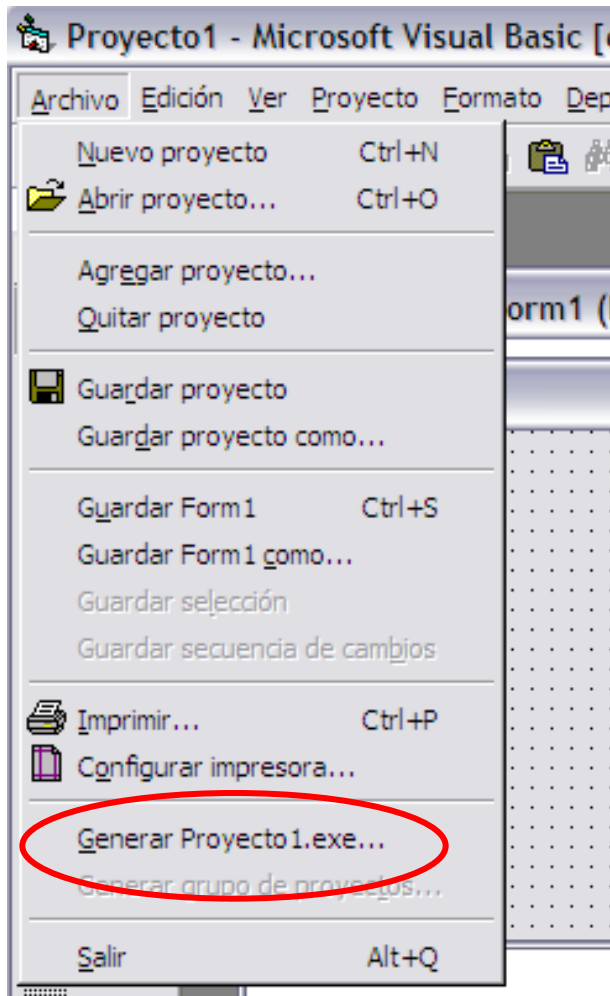
Ejercicio 4



Visual Basic

- Para generar un fichero ejecutable de la aplicación, hay que ejecutar la orden **Generar proyecto.exe** del menú **Archivo**. Para

ejecutar un fichero de este tipo se requiere Windows 95 o posterior y el fichero **msvbvm60.dll**. Dependiendo de los requerimientos de la aplicación, puede ser que necesitemos otros ficheros.



Visual Basic

Ejercicio 5

- Consiste en realizar un pequeño interprete de inglés en el cual, cuando pulsemos el botón “Buenos Días” aparecerá en la caja de texto *Good Morning*, cuando pulsemos el botón “Buenas Tardes” aparecerá en la caja de texto *Good Evening*, y cuando pulsemos el botón “Buenas Noches” aparecerá en la caja de texto *Good Night*. Cuando pulsemos el botón “Red”, la caja se pondrá roja, cuando pulsemos el botón “Blue” la caja se pondrá azul, y cuando pulsemos el botón “Green” la caja se pondrá verde. Cuando pulsemos el botón “Exit” salimos de la aplicación.

Objeto	Propiedad	Valor
Formulario	Caption	Ejercicio 5
	Nombre	Formulario 5
Cuadro de Texto	Text	(En blanco)
	Nombre	Caja
Botón “Rojo”	Caption	Red
	Nombre	BotonRojo
Botón “Verde”	Caption	Green
	Nombre	BotonVerde
Botón “Azul”	Caption	Blue
	Nombre	BotonAzul
Botón “Salir”	Caption	Salir
	Nombre	BotonSalir

Visual Basic

Ejercicio 5

Objeto	Propiedad	Valor
Botón "Buenos Días"	Caption	Buenos Días
	Nombre	BotonBuenosDias
Botón "Buenas Tardes"	Caption	Buenas Tardes
	Nombre	BotonBuenasTardes
Botón "Buenas Noches"	Caption	Buenas Noches
	Nombre	BotonBuenasNoches

Visual Basic

Ejercicio 5

```
Private Sub BotonAZul_Click()  
    Caja.BackColor = vbBlue  
End Sub
```

```
Private Sub BotonBuenasNoches_Click()  
    Caja.Text = "Good Night"  
End Sub
```

```
Private Sub BotonBuenasTardes_Click()  
    Caja.Text = "Good Evening"  
End Sub
```

```
Private Sub BotonBuenosDias_Click()  
    Caja.Text = "Good Morning"  
End Sub
```

```
Private Sub BotonRojo_Click()  
    Caja.BackColor = vbRed  
End Sub
```

```
Private Sub BotonSalir_Click()  
    End  
End Sub
```

```
Private Sub BotonVerde_Click()  
    Caja.BackColor = vbGreen  
End Sub
```

Form1

Good Morning

Buenos Días

Buenas Tardes

Buenas Noches

Red

Blue

Green

Exit