

LECCIÓN 2

En las dos siguientes lecciones vamos a crear una nueva aplicación mediante la cual iremos explicando nuevos objetos y propiedades. Es importante que tú también intentes averiguar para que sirven algunas de las propiedades de estos nuevos objetos que hasta este momento no se hayan explicado.

Crearemos una aplicación que nos permitirá realizar una simple operación matemática entre dos números que introduciremos en dos casillas de texto. Las posibles operaciones a realizar serán la **suma**, la **resta**, la **multiplicación** y por último la **división**. Tendremos una lista en la que podremos ir añadiendo las **operaciones** o las **soluciones** de las operaciones que vamos realizando. La apariencia de nuestra práctica será más o menos la siguiente:

Durante estas lecciones aprenderemos como depurar nuestra aplicación para evitar errores y hacer más fácil el manejo a un posible usuario. Haciendo aparecer, según nos convenga, cuadros de ayuda y mensajes de error.

Propiedades del formulario

En este capítulo vamos a familiarizarnos con algunas de las propiedades más importantes de los formularios, como puede ser la posición en la pantalla del formulario cuando se inicia la aplicación, el color de fondo, los botones de maximizar, minimizar y cerrar, etc.

. Práctica 1

1. Inicia **Visual Basic** y haz lo necesario para que te aparezca un nuevo formulario en pantalla.

Una vez tenemos el formulario en pantalla vamos a cambiarle el tamaño. Recuerda que tienes varias maneras de hacerlo. Utiliza el sistema que tú prefieras. Mira la lección anterior.

2. Pon las propiedades **Height** a **5775** y **Width** a **6045**.

Posición al iniciar la ejecución

3. Haz un clic en **Ventana posición del formulario** del menú **Ver**.



Observa como en algún lugar de la pantalla te ha aparecido una ventana como esta:

Esta ventana nos ofrece una simulación de lo que sería nuestro formulario dentro de la pantalla del ordenador.

4. Sitúate encima del recuadro blanco donde aparece la palabra **Form1**.

Observa como te ha aparecido un cursor, más o menos como este:



Si mantienes pulsado el botón izquierdo del ratón podrás ver como puedes mover el formulario a cualquier parte de la pantalla negra. Con esto conseguimos que el formulario en el momento de ejecutarse se inicie en el lugar que hemos situado el recuadro **Form1**.

5. Coloca el dibujo del formulario en una de las esquinas e inicia una ejecución de prueba. Acto seguido detén la ejecución de prueba.

Observa como el formulario aparece en el lugar de la pantalla que tu le has indicado. Con esta misma pequeña ventana podemos hacer que el formulario, siempre nos aparezca centrado en la pantalla.

*6. Sitúate sobre el dibujo del formulario. Pulsa el botón derecho del ratón para que aparezca el menú contextual. Haz un clic en **Guías de resolución**.*

Con esta opción podrás ver unas guías que te indican como sería la pantalla con resoluciones inferiores a la que tienes actualmente en tu ordenador.

*7. Quita la opción **Guías de resolución** (pulsando otro clic en esta opción) y activa **Centro de la pantalla** dentro de **Posición inicial**.*

Con esta otra opción lo que conseguirás es que el formulario siempre que se ejecute aparezca en el centro de la pantalla del usuario. En nuestra aplicación dejaremos activada esta opción.

Estas mismas opciones las podemos hacer desde la ventana de propiedades dentro de **StartupPosition** con 4 opciones diferentes. **Manual**; **centrado dentro de un formulario padre** (está opción la explicaremos en futuras lecciones); **centrado en la pantalla** o **predefinido por Windows** (Esquina superior izquierda de la pantalla). Si te fijas son las mismas opciones que aparecen dentro del menú contextual al que hemos hecho referencia anteriormente.

Nosotros también podemos modificar la situación del formulario con respecto a los bordes interiores de la pantalla con las propiedades **Top** y **Left**. **Top** nos marca la distancia que existe entre la parte superior del monitor con la parte superior de nuestro formulario, mientras que **Left** nos marca la distancia entre la parte izquierda del monitor y la izquierda de nuestro formulario.

Si te molesta la ventana **Posición del formulario** la puedes cerrar.

Estilo del borde

Con el estilo del borde, **BorderStyle**, lo que podemos conseguir es hacer, por ejemplo, que nuestra aplicación no tenga ningún tipo de borde, que no se pueda cambiar su tamaño, que el tamaño lo podamos variar como a nosotros nos apetezca,...

En nuestro caso nos interesa que no se pueda modificar el tamaño del formulario ya que al hacer más pequeño el formulario se podrían ocultar botones y no podríamos utilizar la aplicación correctamente. Lo que si permitiremos es que el usuario pueda minimizar la aplicación, pero no la pueda maximizar.

Dentro de **BorderStyle** tenemos 6 posibles opciones.

0 - None: Hace que en nuestra aplicación no aparezcan bordes.

1 - Fixed Single: Hace que el borde de la aplicación siempre quede fijo. Con

esta opción podremos poner los botones minimizar o maximizar según nos convenga.

2 - Sizable: Esta opción es la que aparece por defecto al iniciar un nuevo formulario. Con esta opción podemos cambiar el tamaño del formulario a nuestro gusto.

3 - Fixed Double: Con esta opción podemos incluir el menú de control, la barra de título, pero no podemos incluir ni los botones maximizar ni minimizar. Esta ventana no podrá cambiarse de tamaño.

4 - Fixed Tool Window: Si activamos esta opción nos mostrará un formulario con la fuente del título reducida. No podremos modificar el tamaño del formulario. Este no aparecerá en la barra de tareas de Windows.

5 - Sizable Tool Window: Tendremos una ventana de tamaño ajustable. El tamaño de la fuente del título aparecerá reducida. El formulario no aparecerá en la barra de tareas.

Una cosa que hay que tener en cuenta es que estas opciones se ponen en funcionamiento en el momento que ejecutamos la aplicación. Otra cosa a tener en cuenta es que el menú de control que aparece sobre el icono de la aplicación también se modificará según las opciones de **BorderStyle** que hemos seleccionado y los botones de minimizar y maximizar que tengamos activados.

8. Coloca la propiedad **BorderStyle** de nuestro formulario a **1 - Fixed Single**.

Observa como los botones maximizar y minimizar han desaparecido de nuestro formulario, solo queda visible el botón cerrar.

9. Inicia una ejecución de prueba e intenta modificar el tamaño del formulario. Cuando termines detén la ejecución.

Vamos a colocar el botón **minimizar** para que el usuario pueda minimizar el formulario cuando le apetezca. Aunque esté esta opción activada el formulario seguirá sin "dejarse" cambiar el tamaño.

10. Sitúate sobre la propiedad **MinButton**.

Observa como esta propiedad tiene como valor **False**. Esto nos indica que el botón minimizar no está activado.

11. Haz doble clic sobre la palabra **MinButton** y observa como su valor cambia a **True**.

De esta manera hemos hecho que en nuestro formulario aparezca el botón minimizar. Observa como ha aparecido también el botón maximizar pero este no está activado. Para activarlo tendríamos que poner a **True** la propiedad **MaxButton**. En nuestro ejemplo no lo vamos a activar ya que no nos interesa que el usuario pueda maximizar nuestra aplicación.

Si queremos que el usuario no pueda mover por la pantalla la aplicación tendríamos que poner la propiedad **Moveable** a **False**. No es muy recomendado utilizar esta opción, excepto en casos muy específicos, ya que tenemos que dejar que el usuario pueda mover las aplicaciones por la pantalla para así poder visualizar el contenido de otras aplicaciones que están por detrás de esta.

Apariencia del formulario

Vamos a cambiar el texto que aparece en el título del formulario. Recuerda como se hace según lo explicado en la primera lección.

12. Escribe **Pequeña calculadora** como título de nuestra aplicación.

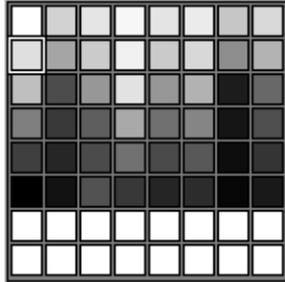
Ahora cambiaremos el icono que aparece en nuestra aplicación.

13. Coloca como icono de la aplicación **Misc18.ico** que se encuentra dentro del direc-

torio **Graphics\Icons\Misc** dentro del directorio donde tengas instalado **Visual Basic**.

Ahora vamos a cambiar el color de fondo de nuestra **Pequeña calculadora**.

14. Accede a la propiedad **BackColor** y haz clic en la flecha para que se despliegue el menú de colores.



Observa que aparecen dos carpetas. Una llamada **Sistema**, donde aparecen los colores de todos los objetos que vienen determinados por Windows y **Paleta** donde nos aparecen diversos colores para elegir. La **Paleta** es parecida a la que mostramos en la derecha.

Si haces clic con el botón izquierdo sobre uno de los cuadrados blancos inferiores te aparecerá una pantalla, en la que podrás elegir un color entre todos los disponibles dentro de la paleta de colores de Windows.

15. Haz clic sobre el color que deseas para el fondo de nuestra aplicación. Yo he seleccionado el gris claro. (Segunda fila, primera columna).

No pongas colores que cansen mucho a la vista ya que debemos pensar que nuestras aplicaciones puede ser utilizada por usuarios durante mucho rato con lo que le puede producir sensación de cansancio.

Añadir objetos al formulario

Vamos a situar en nuestro formulario los elementos que ya conocemos: **Label**, **TextBox** y **CommandButton**. Para ello os mostraremos una lista en la que aparecerá el **tipo** de elemento que deberéis añadir en nuestro formulario, el **texto** que debe aparecer, cual debe ser su **nombre** (en caso de necesitarlo), su **tamaño** y por último su **posición**.

Más adelante modificaremos la apariencia de los objetos que añadiremos ahora.

Repasa la primera lección cuando se explica como se añaden objetos nuevos, como se cambian de tamaño y como se sitúan en una posición determinada dentro del formulario.

16. Deberás añadir 6 objetos **Label**.

Será mejor que los vayas añadiendo y modificando de uno en uno.

Modifica las propiedades de cada **Label** para que queden de la siguiente forma:

Label1

Caption: Calculadora

Left: 1920

Top: 0

Label2**Caption:** Primer operando**Left:** 240**Top:** 1320**Label3****Caption:** Segundo operando**Left:** 2280**Top:** 1320**Label4****Caption:** Resultado**Left:** 4680**Top:** 1320**Label5****Caption:** Lista de operaciones**Left:** 480**Top:** 3360**Label6****Caption:** Operaciones con la lista**Left:** 3360**Top:** 4560**Label7****Caption:** 0**(Nombre):** MostrarResultado**Left:** 4560**Top:** 1560**BorderStyle:** 1 - Fixed Single

Observa que a los 6 primeros **Label** que hemos añadido a nuestro formulario, no le hemos puesto **(Nombre)** a ninguno. Esto es debido a que durante la ejecución de esta aplicación estos objetos no deberán sufrir ningún tipo de modificación con lo que el nombre no nos interesa.

En cambio, el **Label7** será donde nos aparecerá el resultado de la operación que deseamos realizar.

*17. Añade 2 objetos **CommandButton**.*

Modifica las propiedades de cada **CommandButton** para que queden de la siguiente forma:

Command1**Caption:** Borrar**(Nombre):** Borrar**Left:** 3000**Top:** 2400**Width:** 1215**Height:** 495**Command2****Caption:** Calcular

(Nombre): Calcular
Left: 4320
Top: 2400
Width: 1215
Height: 495

Recuerda como activar las teclas del modo abreviado de cada **Command**. Ejemplo: Botón **Calcular Alt+C**.

18. *Añade 2 objetos **TextBox**.*

Modifica las propiedades de cada **TextBox** para que queden de la siguiente forma:

Text1

Text: (Borra el texto actual)
(Nombre): PrimerOperando
Left: 240
Top: 1560

Text2

Text: (Borra el texto actual)
(Nombre): SegundoOperando
Left: 2400
Top: 1560

Observa que en muchos de los objetos que hemos añadido hasta el momento no hemos especificado el tamaño, esto lo haremos más adelante cuando modifiquemos otras nuevas propiedades de estos objetos.

Modificar propiedades de varios objetos simultáneamente

Vamos a modificar una propiedad que tendrán en común varios objetos.

Si varios objetos que tenemos en nuestro formulario cumplen una misma propiedad podemos hacer dos cosas: podríamos ir seleccionando objeto a objeto y modificar la propiedad en cada uno de ellos o seleccionarlos todos y modificar de una sola vez la propiedad con lo que quedarían todos los objetos modificados.

19. *Haz un clic sobre **Calculadora**.*

20. *Pulsa la tecla **Control** y mientras la mantienes pulsada haz clic en **Primer operando**, **Segundo operando**, **Resultado**, **Lista de operaciones** y **Operaciones con la lista**.*

Observa como han quedado seleccionados todos los elementos que hemos marcado. Observa también como la lista de propiedades ha cambiado, solo se muestran las propiedades que podemos cambiar de forma conjunta a todos los objetos seleccionados.

Si te fijas en los objetos seleccionados podrás observar que tienen un fondo de color gris oscuro que delimita su tamaño. (Esto solo lo podrás ver si el color que escogiste para el formulario es diferente a este gris). Lo que vamos a hacer es ver una nueva propiedad que nos hará que estos objetos sean transparentes, de esta manera conseguiremos que solo se vea el texto y no el tamaño de dicho objeto.

21. *Pulsa **F4** para acceder a las propiedades.*

22. *Haz un doble clic sobre la propiedad **BackStyle** verás como todos los objetos selec-*

cionados pasan de ser **opacos** a **transparentes**.

Fuentes de letra en modo edición.

La gran mayoría de los objetos que podemos añadir a un formulario contienen texto. Este texto también puede modificarse para hacer más vistosa o más clara nuestra aplicación. El formato de texto se puede cambiar desde el modo diseño o desde el modo de ejecución (como ya vimos en la lección anterior). En este capítulo explicaremos con más detenimiento ambos sistemas.

23. Si todavía mantienes seleccionados los objetos que hemos seleccionado en los anteriores puntos sólo debes hacer un clic, manteniendo pulsada la tecla **Control**, sobre **Calculadora**, para quitar la selección de este objeto. Si no mantienes la selección, vuelve a seleccionar los objetos que antes teníamos seleccionados pero esta vez sin el texto **Calculadora**.

Esto lo hemos hecho porque todos los objetos que están seleccionados tienen el mismo formato de letra mientras que el título **Calculadora** tiene otro formato.

24. Pulsa **F4**.

25. Accede a la propiedad **Font**.

Observa que esta propiedad está vacía. Esto siempre ocurre en el momento en el que tenemos diferentes objetos seleccionados.

26. Pulsa un clic sobre el botón con tres puntos suspensivos que aparece en dicha propiedad.

Acto seguido aparece un cuadro de diálogo como este:

Dentro de la lista **Fuente** podremos seleccionar uno de los tipos de letras que tenemos instalado en nuestro ordenador. En el apartado **Estilo de fuente** podremos seleccionar entre cuatro opciones **Normal** (ejemplo), **Cursiva** (ejemplo), **Negrita** (ejemplo), **Negrita cursiva** (ejemplo). Juntamente con el estilo seleccionado podemos aplicar dos **Efectos** diferentes como es: **Tachado** (ejemplo) o **Subrayado** (ejemplo). También podemos hacer una mezcla de los diferentes formatos de letra para así poder obtener algo así (ejemplo): negrita cursiva con subrayado y tachado. También podremos modificar el **Tamaño** de la fuente seleccionada. Debemos tener cuidado con esta propiedad ya que según el tamaño que seleccionemos podría ser que no se viera completamente el contenido de la información que deseamos mostrar.

27. Haz un clic en **Negrita**. Acepta la ventana.

Observa los cambios. Las demás opciones las dejaremos como están. Si no ves todo el contenido de estos elementos, no pasa nada.

28. Haz un clic en cualquier parte de la pantalla para quitar la selección.

29. Haz un clic sobre **Calculadora**.

30. Accede a la propiedad **Font**.

31. Accede al cuadro de diálogo **Fuente**.

32. Modifica el **tamaño** a **18** y haz que aparezca **Subrayado**.

Antes de aceptar la ventana observa el recuadro de **Ejemplo**. En este recuadro podrás ver una simulación de cómo quedarán las modificaciones que has hecho.

33. **Acepta** el cuadro de diálogo.

34. Selecciona el **TextBox** que lleva como **(Nombre) PrimerOperando** y **SegundoOperando**, junto con el **Label** llamado **MostrarResultado**.

35. Accede al cuadro de diálogo **Fuente**.

36. Cambia el **Tamaño** a **18**. Acepta el cuadro de diálogo.

Modificar tamaños

Vamos a modificar el tamaño de estos 3 últimos objetos modificados.

37. Modifica el tamaño a **1215 x 540**

Fuentes de letra en modo ejecución

Como ya vimos en la lección anterior los estilos de fuente se pueden modificar mientras estamos ejecutando el programa. Esto se consigue modificando las propiedades de **estilo de fuente** de alguno de los objetos insertados en nuestro formulario.

Vamos a imaginarnos que tenemos un objeto llamado **Texto** en nuestro formulario de trabajo al cual le modificaremos los estilos de fuente.

Para modificar un estilo de fuente como puede ser **negrita**, **cursiva**, **tachado** y **subrayado** utilizamos unas propiedades de tipo **booleano**¹. Su sintaxis es exactamente igual que en el caso de cualquier otra propiedad. Debemos escribir el nombre del objeto que queremos modificar seguido de un punto y una de estas cuatro propiedades: **FontBold** (**Negrita**), **FontItalic** (*Itálica*), **FontStrikethru** (~~Tachado~~) o **FontUnderline** (Subrayado), después el signo igual (=) y el valor **True** o **False** según nos interese activarlo o desactivarlo. (También podríamos poner **Font.Bold**, **Font.Italic**, **Font.Strikethru** o **Font.Underline**).

Por ejemplo, imaginemos que tenemos un **botón** que al pulsarlo queremos que el objeto **Texto** cambie a **negrita**. Dentro del objeto **botón** escribiremos la siguiente línea de código **Texto.FontBold = True** esto hará que el **Texto** aparezca en **negrita**. Si ya está en **negrita** no ocurrirá nada. Si queremos que aparezca el texto "normal" podríamos poner en otro **botón** la línea **Texto.FontBold = False**, de esta manera tendremos un **botón** que activa la negrita y otro que la desactiva. Esto funciona exactamente igual para cualquiera de las otras propiedades.

Si te fijas en este caso tenemos que diseñar dos botones para activar y desactivar la negrita, pero podemos hacer que un mismo botón haga las dos cosas, o cualquier otra propiedad, según la que exista en este momento. Lo explicaremos de otra forma; si el texto está en negrita se desactivará la negrita y si el texto no está en negrita se activará la negrita. Esto se consigue con la siguiente línea: **Texto.FontBold = Not Texto.FontBold**. La partícula **Not** hace que la propiedad se alterne, si está en **False** se convierte en lo contrario **True** y si su valor es **True** se convierte en **False**.

También podemos cambiar el tipo de fuente, esto lo haremos con la propiedad **FontName**. Esta propiedad no es de tipo **Booleana** ya que tenemos que especificar el nombre de la fuente que queremos insertar. La sintaxis sería de la siguiente forma: Nombre del objeto seguido de un punto, la propiedad **FontName**, un igual y entre comillas dobles el nombre de la fuente. Por ejemplo: **Texto.FontName = «Verdana»**.

Otra propiedad que tenemos para cambiar nuestro estilo de fuente es: **FontSize**, con esta propiedad lo que conseguimos es modificar el tamaño de la fuente. Esta propiedad tampoco es de tipo **booleana** ya que deberemos especificar el tamaño de la fuente. El tamaño se expresa en puntos. El tamaño máximo es de **2160 puntos**. Los puntos son de tipo numérico con lo que la sintaxis sería de la siguiente manera: Nombre del objeto seguido de un punto, la propiedad **FontSize**, un igual y el número que

indicará el tamaño de la fuente de letra. Por ejemplo: **Texto.FontSize = 12**.

Como practica adicional puedes crear un nuevo formulario para practicar estas nuevas propiedades.

Tamaño automático

Ahora vamos a modificar el tamaño de los cuadros de texto que tenemos en nuestro formulario.

Emplearemos otra nueva propiedad de estos objetos que es el ajuste automático del tamaño con respecto al texto que hay en su interior.

38. *Selecciona todos los elementos de texto que tenemos hasta el momento, menos el que tiene como **(Nombre) MostrarResultado**.*

39. *Accede a las propiedades y cambia a **True** la propiedad **AutoSize**.*

Observa como los puntos de selección de cada uno de los objetos se ha aproximado hasta el texto. Si nosotros ahora modificásemos la propiedad **Caption** veríamos como el tamaño del objeto cambia según el tamaño del texto que hay dentro de dicho objeto.

Alineación del texto

En nuestra práctica nosotros vamos a trabajar con diferentes números que iremos introduciendo en las casillas de primer y segundo operando para obtener un resultado.

Si nosotros utilizamos casillas de texto o etiquetas para que el usuario introduzca o visualice texto, normalmente se alinea a la izquierda (ya que es por donde comenzamos a escribir texto) y si trabajamos con números los alineamos a la derecha (para que todas las comas decimales en los números enteros estén juntas).

40. *Selecciona solo **PrimerOperando** y accede a la propiedad **Text**.*

41. *Escribe la palabra **Texto**.*

Esto lo hemos hecho para poder explicar mejor como actúa la alineación del texto en los diferentes objetos.

Observa como en este objeto al igual que en el **Label MostrarResultado** el texto está a la izquierda.

42. *Accede a la propiedad **Alignment** de **MostrarResultado**.*

Observa que tienes 3 posibles opciones. Esto lo podrás ver si despliegas la lista de esta propiedad. **0: izquierda, 1: derecha, 2: centro**.

43. *Selecciona la alineación a la derecha (**1.- Right Justify**).*

Observa nuestro formulario y donde está alineado el texto de este objeto.

Vamos a hacer lo mismo con los objetos: **PrimerOperando** y **SegundoOperando**. Si quieres ver mejor los cambios y para asegurarte que lo haces correctamente puedes poner algo en la propiedad **Text** de **SegundoOperando**.

44. *Selecciona **PrimerOperando** y **SegundoOperando** para trabajar con ambos objetos conjuntamente.*

45. Accede a la propiedad **Alignment** y selecciona la opción correspondiente, para hacer que el texto de estos objetos aparezca alineado a la derecha.

Observa como en los dos objetos de tipo texto que tenemos seleccionados no ha ocurrido absolutamente nada. ¿A que es debido este comportamiento? Muy sencillo, si queremos que esta propiedad "funcione" tenemos que activar otra propiedad.

46. Accede a la propiedad **MultiLine** y ponla en **True**.

Observa que inmediatamente después de cambiar esta opción el texto pasa a estar alineado a la derecha.

La propiedad **MultiLine** lo que está haciendo es definir que en los dos objetos texto se puedan introducir varias líneas. Ten en cuenta que siempre que quieras una alineación a derecha o centro en objetos **Text** deberás activar la propiedad **MultiLine**.

47. Borra el contenido de los dos objetos seleccionados.

Observa que no podrás modificar el contenido de los objetos mientras estén los dos seleccionados.

Al acceder a la propiedad **Text** verás que hay la palabra (**Texto**) esto nos indica que **MultiLine** está activado y por lo tanto puede ser que dentro de este objeto puedan existir múltiples líneas de texto. Para eliminar lo que ya tenemos debemos pulsar en el botón con una flecha hacia abajo que aparece en esta propiedad y borrar el contenido.

Delimitación de tamaño

Ahora vamos a delimitar el tamaño de los números que podemos introducir en **PrimerOperando** y **SegundoOperando**. Esto lo conseguiremos con la propiedad **MaxLength**. Esta propiedad hará que no podamos introducir números con una cantidad de caracteres superiores a la que nosotros indiquemos. **Visual Basic** no nos dejará introducir más caracteres. No nos avisará de ninguna manera, simplemente no nos dejará introducir ningún carácter más.

48. Selecciona **PrimerOperando** y **SegundoOperando**.

49. Pulsa **F4**, para acceder a las propiedades.

50. Escribe **4** en **MaxLength**.

Texto de ayuda

Existe una propiedad, en la mayoría de los objetos que podemos añadir en nuestro formulario, que sirve para mostrar **ayuda** rápida al mantener el puntero del ratón durante unos segundos sobre el objeto deseado. Este texto suele ser corto y explícito dando una idea de para que sirve dicho control.

51. Selecciona **PrimerOperando**.

52. Accede a sus propiedades.

53. Sitúate sobre la propiedad: **ToolTipText**.

En esta propiedad podemos escribir lo que queremos que aparezca en el pequeño cuadro de ayuda al mantener el ratón durante unos segundos en el objeto seleccionado.

54. Escribe: **Introduce el primer operando**.

Realiza estas mismas operaciones con **SegundoOperando** y los botones **Calcular** y **Borrar**. Escribe el texto que creas conveniente, pensando que con el botón **Calcular** se realizarán los cálculos pertinentes según la operación seleccionada (opciones que veremos en la siguiente lección) y el botón **Borrar** borra el contenido de **PrimerOperando**, **SegundoOperando** y **MostrarResultado**, para poder iniciar una nueva operación con diferentes operandos.

OptionButton en modo gráfico

Vamos a insertar unos controles que nos servirán para poder seleccionar cual de las cuatro operaciones (suma, resta, multiplicación o división) es la que deseamos realizar. Hemos escogido este elemento ya que solo podremos marcar uno de ellos a la vez.

En la primera lección ya utilizamos este tipo de objeto, pero aquí vamos a ver una nueva propiedad de este, ya que no trabajaremos con él con la apariencia que lo hicimos en la pasada lección, sino que tendrá apariencia de botón, pero con una imagen en su interior.

55. Inserta un **OptionButton**.

Observa como es su apariencia.

56. Ponle como **(Nombre)**: **Sumar**.

57. Accede a la propiedad **Style** y modifica su valor de **Standard** a **Graphical**.

Observa como su apariencia ahora es como un botón.

58. Borra el contenido de la propiedad **Caption**.

59. Accede a la propiedad **Picture** y selecciona **Misc18.ico** de **Graphics\Icons\Misc** dentro del directorio donde tengas instalado **Visual Basic**.

60. Cambia el tamaño a **540 x 540** y su posición a **1680, 600**.

61. Inserta 3 **OptionButton** más.

63. Modifica sus propiedades para que queden de la siguiente manera:

Option1

Caption: (Borra su contenido)

(Nombre): Restar

Posición: 1680, 1200

Tamaño: 540 x 540

Style: Graphical

Picture: Misc19.ico

Option2

Caption: (Borra su contenido)

(Nombre): Multiplicar

Posición: 1680, 1800

Tamaño: 540 x 540

Style: Graphical

Picture: Misc20.ico

Option3

Caption: (Borra su contenido)

(Nombre): Dividir

Posición: 1680, 2400

Tamaño: 540 x 540

Style: Graphical

Picture: Misc21.ico

64. Asegúrate que la propiedad **Value** de objeto **Sumar** está en **True**.

65. Escribe en la propiedad **ToolTipText** de cada uno de estos objetos algo que le pueda servir de ayuda a los usuarios de esta aplicación, tal como vimos en puntos anteriores.

66. Realiza una ejecución de prueba. Selecciona las diferentes operaciones.

Observa que cuando se selecciona una, se quita la selección la que estaba seleccionada y así sucesivamente.

Creación de archivos ejecutables

Con **Visual Basic** podemos crear archivos ejecutables, con extensión (.exe) de una forma fácil y sencilla.

Al convertir una aplicación en un archivo ejecutable podremos poner en funcionamiento dicha aplicación sin necesidad de ejecutar **Visual Basic**. Pero deberemos pensar que esta aplicación no podrá funcionar en cualquier otro ordenador, solamente funcionará en los aquellos en los que esté instalado **Visual Basic**. Para que funcione en cualquier otro ordenador deberemos hacer un archivo de instalación, más adelante veremos como hacerlo.

Antes de crear el archivo ejecutable lo que tendremos que hacer es guardar el formulario y el proyecto con el nombre que quieras dentro del directorio que quieras.

Antes de continuar, decir que la aplicación no está finalizada pero ya podemos crear el archivo ejecutable para tener una muestra.

Otra cosa muy importante es ir grabando el formulario y el proyecto a medida que vamos haciendo cambios para hacer copias de seguridad. Esto lo podemos hacer de una forma fácil, una vez grabados ya el formulario y el proyecto con un nombre, deberemos pulsar el botón con el dibujo de un disco en la barra de herramientas estándar.

67. Accede a **Generar (nombre del archivo) .exe...** dentro del menú **Archivo**.

En **(nombre del archivo)** te aparecerá el nombre del proyecto que hayas puesto anteriormente.

Seguidamente te aparecerá una ventana llamada **Generar proyecto**. En esta ventana deberemos indicar la carpeta y el nombre del archivo ejecutable.

68. Deja el nombre que aparece en la ventana **Generar proyecto**, antes de pulsar **Aceptar** haz clic en el botón **Opciones**.

Aparecerá un nuevo cuadro de diálogo.

En este cuadro podemos modificar el número de versión de nuestro archivo ejecutable de forma manual, indicando el **número de versión** en **principal**, **secundario** y **revisión**. También lo podemos hacer de forma automática pulsando un clic en **Incremento automático**. Desde este cuadro de diálogo podemos modificar tanto el **título** de nuestra aplicación, como el **icono**. Podemos añadir información adicional como el **nombre de la compañía**, **nombre del producto**, **copyright**, etc.

69. Cierra el cuadro **Propiedades del proyecto**. Haciendo clic en **Cancelar** o en **Aceptar** si has modificado alguna opción.

70. Haz clic en **Aceptar**, para crear el archivo ejecutable.

Observa como en la barra de herramientas estándar aparece una banda de color que va incrementando con la palabra **Compilando...** y **Escribiendo Exe...** Esto nos indica que **Visual Basic** antes de crear el ejecutable compila el proyecto para revisar errores y después graba el archivo con extensión **Exe**.

Para ejecutar el archivo **Exe** que acabamos de crear puedes hacerlo como cualquier otro programa que tengas instalado en tu ordenador.

71. Abre el **Explorador de Windows**, accede al directorio donde está el archivo ejecutable.

Observa como este archivo tiene como icono el mismo que pusimos en el formulario.

72. Haz doble clic para ejecutarlo.

Observa como aparentemente la ejecución es igual que si la hubiéramos hecho desde **Visual Basic**. Podrás observar que en según que tipo de aplicación será mucho más rápido ejecutar el archivo **Exe** que ejecutarlo desde **Visual Basic**.

Vamos a explicar como podemos hacer un archivo de instalación. Recuerda que la aplicación no está finalizada. Este paso normalmente se hace en el momento en el que ya está la aplicación completamente terminada.

Archivo de instalación

Para realizar estos pasos deberemos poner en funcionamiento uno de los módulos que vienen junto a **Visual Basic**.

73. Cierra **Visual Basic**.

Si al realizar este paso, **Visual Basic** te pide si deseas guardar la aplicación contesta afirmativamente.

74. Accede a **Inicio – Programas - Microsoft Visual Studio 6.0 - Herramientas de Microsoft Visual Studio 6.0 - Asistente para empaquetado y distribución**.

Seguidamente te aparecerá una pantalla como la siguiente:

75. Pulsa el botón **Examinar...**

En la ventana **Abrir proyecto** escoge el proyecto con el que hemos estado trabajando anteriormente.

76. Cuando lo tengas seleccionado pulsa el botón **Abrir**.

Observa como en la lista desplegable de la parte superior de la ventana **Asistente de empaquetado y distribución** aparece la carpeta y el nombre del archivo con el que deseas trabajar.

77. Pulsa el botón **Empaquetar**.

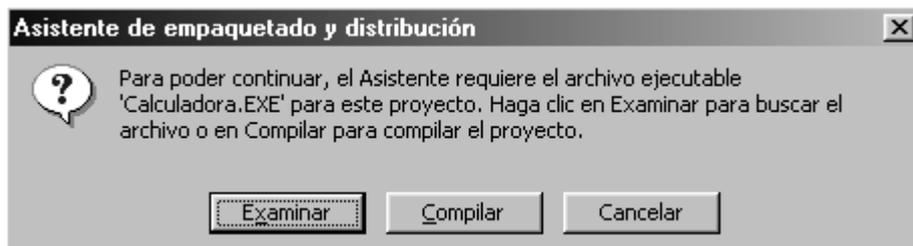
En este momento **Visual Basic** comprobará si tenemos realizado el archivo ejecutable (exe), si no lo hemos creado anteriormente nos aparecerá una ventana parecida a la siguiente:



En la que deberemos seleccionar la opción **Compilar**. De esta forma **Visual Basic** generará el archivo ejecutable.

78. Pulsa en botón **Compilar**.

En caso de ya estar creado dicho archivo no nos aparecerá esta ventana.



Seguidamente nos aparecerá una nueva ventana como esta:



79. Selecciona la opción **Paquete de instalación estándar** y pulsa en **Siguiente**.

Al elegir esta opción estamos indicando que se genere un archivo de instalación para nuestra aplicación.

Después de unos segundos nos aparecerá la siguiente pantalla:



En ella deberemos indicar en que carpeta deseamos guardar los archivos que se generarán para realizar la instalación de nuestra aplicación. Observa que él nos propone utilizar una carpeta nueva llamada **Paquete** que cuelga de la carpeta en la que tenemos nuestro proyecto.

80. Vamos a dejar esta carpeta como buena. Pulsa en **Siguiente**.

Si el asistente ve que la carpeta no existe nos mostrará un mensaje para crearla.

81. Responde afirmativamente a este mensaje.

Seguidamente nos aparecerá una nueva ventana en la que se nos mostrarán todos los archivos necesarios para que funcione perfectamente nuestra aplicación cuando se instale en cualquier ordenador. Nosotros podremos indicar que no se instale un archivo de los que aparecen en la lista, el único problema que puede haber es que según el archivo que quitemos no funcione correctamente la aplicación en cualquier ordenador. También podemos hacer que se guarde algún otro archivo junto con la instalación. Para ello deberíamos pulsar sobre el botón **Agregar...**

La pantalla tiene un aspecto parecido a este:



Vamos a dejar seleccionados todos los archivos y no vamos a **Agregar** ningún archivo más.

82. Pulsa en **Siguiete**.

Seguidamente nos aparecerá una pantalla como la siguiente:



En esta pantalla podremos indicar si queremos que se genere un único archivo **.cab** o varios archivos más pequeños **.cab** para así poderlos guardar en discos. Los archivos **.cab** son los que contienen toda la información y todos los archivos que se descomprimen durante la instalación.

En nuestro caso vamos a generar un sólo archivo **.cab**

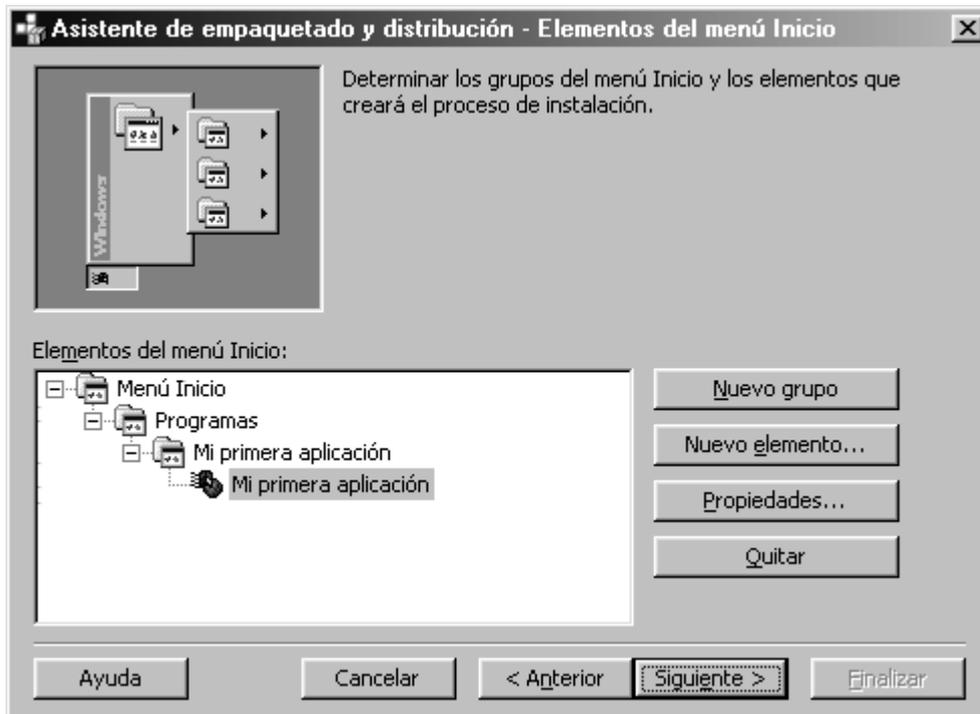
83. Selecciona la opción: **Un único archivo .cab** y pulsa en **Siguiete**.

En la siguiente pantalla que nos aparece de todo este proceso deberemos indi-

car el título que deseamos que aparezca en el momento en el que se realiza la instalación.

84. Escribe: **Mi primera aplicación** y pulsa en **Siguiente**.

En esta nueva ventana:



Podremos indicar en que grupo del menú **Inicio** deseamos que se guarde el acceso directo hacia nuestra aplicación.

Vamos a cambiar el nombre del grupo en el que estará situado el acceso directo a nuestra aplicación.

85. Pulsa sobre el grupo **Mi primera aplicación**.

86. Haz clic sobre el botón **Propiedades...**

87. En la ventana que te aparecerá a continuación cambia el nombre y escribe: **Mis aplicaciones**.

88. Pulsa sobre el botón **Aceptar**.

Una vez cambiado el nombre del grupo ya podemos seguir con nuestro asistente.

89. Pulsa sobre el botón **Siguiente**.

En la siguiente ventana podríamos modificar en que carpetas queremos que se guarde la aplicación en el momento que se instale.

Vamos a dejar la configuración original y así permitiremos que sea el usuario

quien decida en que carpeta desea guardar la aplicación en el momento en el que se realice la instalación.

90. *Pulsa sobre el botón **Siguiente**.*

Seguidamente nos aparecerá una pantalla en la que podremos indicar si el archivo ejecutable que se generará después de realizar la instalación deseamos que esté compartido o no. El estar compartido quiere decir si este programa formará parte de otros o solamente será este el que lo utilice. Así en el momento de desinstalar la aplicación no se borrará dicho archivo para que pueda ser utilizado por otros programas.

91. *No selecciones este archivo y pulsa sobre el botón **Siguiente**.*

En la ventana que nos aparece a continuación podemos indicar un nombre para guardar gran parte de los pasos que hemos realizado durante este asistente para que así en futuras ocasiones podamos realizar dichos pasos automáticamente.

92. *Deja el nombre que aparece por defecto y pulsa sobre el botón **Finalizar**.*

Este es el final del proceso. Ahora se generarán todos los archivos necesarios para realizar la instalación de la aplicación.

Al final del proceso nos aparecerá un cuadro con un pequeño informe del proceso realizado.

93. *Pulsa en el botón **Cerrar**.*

94. *Por último cierra el **Asistente de empaquetado y distribución**.*

Ahora ya tienes la aplicación preparada para grabársela a quien desees.

95. *Abre el explorador, accede a la carpeta donde has guardado los archivos de instalación y ejecuta el archivo **Setup.exe**.*

96. *Sigue los pasos, así podrás observar como se realiza la instalación de la aplicación.*

La aplicación ya tiene un aspecto mucho más serio y profesional.

Fin lección 2

¹ Booleano: solo puede tomar dos valores, Verdadero o Falso.