

Resumen VB .Net

- La plataforma .NET de Microsoft es un componente de software que puede ser añadido al sistema operativo Windows. Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma. Esta solución es el producto principal en la oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows.
- La POO es una técnica para desarrollar soluciones computacionales utilizando componentes de *software* (objetos de *software*).

Palabras Reservadas

PALABRAS RESERVADAS DEL LENGUAJE			
AddHandler	AddressOf	Alias	And
AndAlso	As	Boolean	ByRef
Byte	ByVal	Call	Case
Catch	CBool	CByte	CChar
CDate	CDec	CDbl	Char
CInt	Class	CLng	CObj
Const	Continue	CSByte	CShort
CSng	CStr	CType	CUInt
CULng	CUShort	Date	Decimal
Declare	Default	Delegate	Dim
DirectCast	Do	Double	Each
Else	Elseif	End	EndIf
Enum	Erase	Error	Event
Exit	False	Finally	For
Friend	Function	Get	GetType
GetXMLNamespace	Global	GoSub	GoTo
Handles	If	If()	Implements
Imports (.NET)	Imports (XML)	In	Inherits
Integer	Interface	Is	IsNot
Let	Lib	Like	Long
Loop	Me	Mod	Module

PALABRAS RESERVADAS DEL LENGUAJE			
MustInherit	MustOverride	MyBase	MyClass
Namespace	Narrowing	New	Next
Not	Nothing	NotInheritable	NotOverridable
Object	Of	On	Operator
Option	Optional	Or	OrElse
Overloads	Overridable	Overrides	ParamArray
Partial	Private	Property	Protected
Public	RaiseEvent	ReadOnly	ReDim
REM	RemoveHandler	Resume	Return
SByte	Select	Set	Shadows
Shared	Short	Single	Static
Step	Stop	String	Structure
Sub	SyncLock	Then	Throw
To	True	Try	TryCast
TypeOf	Variant	Wend	UInteger
ULong	UShort	Using	When
While	Widening	With	WithEvents
WriteOnly	Xor	#Const	#Else
#Elseif	#End	#If	=
&	&=	*	*=
/	/=	\	\=
^	^=	+	+=
-	-=	>> (Operador)	>>= (Operador)
<<	<<=		

Tipos de Datos

Tipo	Valores	Tamaño
Boolean	Representa un valor verdadero (true) o falso (false)	2 bytes
Byte	Representa un valor de 8 bits en un rango entre 0 y 255	1 byte
Char	Representa un carácter de 16 bits	2 bytes
DateTime	Representa un valor de fecha y hora	8 bytes
Decimal	Representa un valor de 28 dígitos significativos	12 bytes
Double	Representa un valor en coma flotante de 64 bits	8 bytes
Integer	Representa un valor entre un rango de +- 2,147,483,698	4 Bytes
Long	Representa un valor entre un rango de +- 9.223.372.036.854.775.807	8 Bytes
Short	Representa un valor entre un rango de +- 32.677	2 Bytes
String	Cadena de caracteres	0 a 2 billones de caracteres

Variables/Constantes: Espacio de memoria que permite almacenar un tipo de dato

Variables Locales

Dim <nombre de variable> **as** <Tipo de Dato>

Variables Globales

Global<nombre de variable> **as** <Tipo de Dato>

Constantes

Variables a las que solamente se les puede asignar una vez el valor y permanece constante a lo largo del programa

Const <nombre de variable> = <valor constante>

Operadores Aritméticos

Operadores de Visual Basic .NET	Operador	<u>Expresión</u>
Suma	+	numero1 +numero2
Resta	-	numero1 – numero2
Multiplicación	*	Numero1 * numero2
División	/	Numero1 / numero2
División entera	\	Numero1 \ numero2
Residuo	mod	numero1 mod numero2
Exponenciación	^	Numero1 ^ numero2

Operadores Relacionales

Operadores Relacionales	Operador	Expresión
Mayor que	>	variable1>variable2
Mayor o igual que	>=	variable1>=variable2
Menor que	<	variable1<variable2
Menor o igual que	<=	variable1<=variable2
Diferente	<>	variable1<>variable2
Igual a	=	variable1=variable2

Operadores Lógicos

Operadores lógicos	Operador	Expresión
Y	And	Es verdadero, si al evaluar cada uno de los operandos el resultado es verdadero, si uno de los operandos es falso el resultado será falso.
También Y	AndAlso	Es falso, si al evaluar el primer operando el resultado es falso, el segundo operando no es evaluado.
O	Or	Es falso, si al evaluar cada uno de los operandos el resultado es falso, si uno de los operandos es verdadero el resultado será verdadero.
También O	OrElse	Es verdadero, si al evaluar el primer operando el resultado es verdadero, el segundo operando no es evaluado.
Negación	Not	El resultado de aplicar este operando es falso si al evaluar su operando el resultado es verdadero, y verdadero en caso contrario.
	Xor	Da como resultado verdadero, si al evaluar cada uno de los operando uno de ellos es verdadero y el otro falso, caso contrario es falso.

Sentencias de control

If (condición) Then

Instrucción(es) a ejecutarse si la condición es verdadera

Else

Instrucción(es) a ejecutarse si la condición es falsa

Endif

```
While (condición)
    Instruccion(es)
End While
Resto del programa
```

```
Do
    Instrucción(es)
Loop While (condición)
```

```
For variable =expresion1 To expresión2 Step expresion3
    Instruccion(es)
Next
```

```
Select (variable)
    Case expresion1
        instrucciones1
    Case expresion2
        instrucciones2
    ...
    ...
    Case Else:
        instruccionesN
EndSelect
```

```
Case Is <y      ' variable < y
Case 3          ' variable =3
Case y to 10    ' variable = y, y+1,.....,10
Case 3, x       ' variable = 3, x
Case -5, w To 5  ' variable = -1, w, w+1.....,5
Case "dato", "DATO" ' variable ="dato", "DATO"
Case Is >=200   ' variable >=200
```

Entorno de Desarrollo

1. Menús: agrupa las funciones de personalización del IDE, la ayuda, la visualización de las ventanas contenedoras de herramientas, el acceso a propiedades, y la función de compilación de proyectos.
2. Barra de herramientas: en ella se pueden agrupar los principales botones equivalentes a las funciones más utilizadas de los menús. La barra principal incluye las funciones de iniciar un nuevo proyecto, guardar, agregar uno o más objetos al proyecto, compilar y depurar, entre otras.

3. Cuadro de herramientas: agrupa los controles que permiten diseñar la interfaz gráfica de las aplicaciones (botones, cajas de texto, grilla de datos, etc.). Algunos controles pueden no estar visibles al momento de ejecutar nuestro software.
4. Editor de código: es el área de trabajo donde escribimos las sentencias que ejecutará el programa para funcionar.
5. Explorador de soluciones: en él se listarán todos los archivos y recursos internos y externos que componen un proyecto.
6. Ventana de propiedades: desde ella podemos asignar el nombre a formularios y controles, y ajustar las propiedades de cada uno de ellos, la posición en pantalla, las fuentes y los colores, entre otras características.
7. Ventana de inmediato: aquí veremos en modo depuración los valores asignados a variables o a las propiedades de nuestros controles y objetos.
8. Lista de errores: la lista de errores, advertencias y mensajes nos mantendrá informados sobre las equivocaciones cometidas al escribir el código y que el motor de depuración detecte. También permite evaluar datos de variables o espacios de nombre no declarados, entre otras advertencias.

CONTROLES

Los **controles** son las herramientas gráficas que utilizamos para construir la interfaz de usuario de la mayoría de las aplicaciones realizadas en el entorno de programación.

Controles comunes

Los **controles comunes** son los que habitualmente encontramos en cualquier aplicación Windows. No todos son iguales ni cumplen una función **visible** dentro de los programas. Algunos de ellos pueden ser **invisibles**, o su forma inicial en **modo Desarrollo** puede ser distinta de la que toman en el **modo Ejecución** de las aplicaciones.

Eventos de cada control

La mayoría de los controles poseen eventos. Los eventos contienen porciones de código o algoritmos que se ejecutan ante una determinada acción del usuario que utiliza la aplicación o de una acción que realice el sistema operativo.

Button

Representa los botones en las aplicaciones.

EVENTOS	DESCRIPCIÓN
Click, Enter, MouseUp, MouseDown, MouseMove, MouseHover	Son los eventos más relevantes de este control. En el nombre de cada uno se describe la acción que se está realizando.
KeyPress, KeyUp, KeyDown	Estos eventos permiten detectar la información proporcionada por el teclado.

Label, LinkLabel

Estos controles permiten mostrar un texto. **LinkLabel** se diferencia de **Label** porque el texto mostrado se hace como si fuera un **hipervínculo**, con lo cual nos permite indicar al igual que en un website, que haciendo **click** en él y con código

mediante, iremos hacia otro **formulario** o **página web**.

PROPIEDAD	DESCRIPCIÓN
Text	Permite especificar el texto que mostrará.
TextAlign	Permite alinear el texto contenido.
Autosize	Permite tomar un tamaño que se ajuste o no al texto contenido.
Image	Permite mostrar una imagen.
Font	Permite especificar una fuente, tamaño y estilo con el que se mostrará el texto.

TextBox

Este control permite el ingreso de datos mediante el teclado.

PROPIEDAD	DESCRIPCIÓN
Text	Texto que mostrará.
TextAlign	Permite alinear el texto contenido.
MaxLength	Se puede indicar la cantidad máxima de caracteres. El valor 0 permite ingresar texto ilimitado.
CharacterCasing	Procesa el texto ingresado, para transformarlo en mayúscula, minúscula, o dejarlo tal como fue escrito.
PasswordChar	Permite especificar un carácter que se visualizará en vez del texto ingresado, como en los casos de ingreso de contraseña.
Multiline	Estableciendo esta propiedad en True, podremos escribir texto en varias líneas.
ScrollBars	Se aplica si la propiedad Multiline es True para mostrar barras de scroll. Puede ser Scroll vertical, horizontal o ambos.
ReadOnly	En True, no permitirá modificar el texto que muestra.
TextChanged	Evento que se produce cuando el texto contenido cambia.
KeyPress	Evento que recibe como parámetro adicional el código de tecla presionada, para realizar una acción.
KeyUp / KeyDown	Similar al evento KeyPress, pero el primero detecta cuando la tecla se soltó, mientras que el segundo detecta cuando se la presionó.

ListBox y ComboBox

Ambos controles funcionan de manera similar en cuanto a sus propiedades y eventos.

ComboBox es una caja de texto desplegable que permite mostrar y seleccionar un ítem entre todos los ingresados.

ListBox es similar a él, aunque muestra los ítem de manera consecutiva.

PROPIEDAD	DESCRIPCIÓN
COMBOBOX	
DropDownStyle	Indica qué estilo de selección mostrará. El más común es DropDownList.
Items	Agrupar dentro de una colección la cantidad de elementos que contiene.
SelectedIndex	Permite obtener o indicar qué elemento de la colección de ítem está seleccionado, y devuelve su número de índice.
Sorted	Permite establecer un ordenamiento de los elementos.
SelectedItem	El elemento seleccionado por defecto.
Items	Colección que contiene los elementos que el usuario puede ver y seleccionar.
SelectionMode	Permite seleccionar uno o más elementos.
MultiColumn	Permite establecer un listBox con más de una columna de datos.
SelectedItems	Permite obtener qué elementos fueron seleccionados por el usuario, solo si SelectionMode se estableció como Múltiple.

DateTimePicker y MonthCalendar

Estos controles permiten manipular datos a través de las fechas. **DateTimePicker** funciona como un **ComboBox** mostrando un calendario, mientras que **MonthCalendar** muestra un calendario completo en pantalla. **DateTimePicker**, como su nombre lo indica, entrega como valor la fecha y hora, mientras que **MonthCalendar** brinda solamente la fecha seleccionada.

PROPIEDADES DATETIMEPICKER	
PROPIEDAD	DESCRIPCIÓN
Value	Permite obtener o indicar un valor fecha del calendario.
MinDate	Especifica cuál es la fecha mínima que se puede mostrar.
MaxDate	Especifica cuál es la fecha máxima que se puede mostrar.
Format	Establece cuál es el formato por mostrar en la caja de texto del control.

PROPIEDADES MONTHCALENDAR	
PROPIEDAD	DESCRIPCIÓN
SelectionRange	Devuelve los valores mínimos y máximos de un rango de fechas seleccionado en el calendario.
MinDate	Misma función que DateTimePicker.
MaxDate	Misma función que DateTimePicker.
CalendarDimensions	Establece los meses que mostrará el calendario.

RadioButton y CheckBox

Estos controles básicos permiten que los usuarios establezcan una selección de opciones. **CheckBox** puede interactuar con otros controles similares a sí mismo, y admite marcar varios al mismo tiempo. **RadioButton** permite marcar uno solo entre varios de su tipo, siempre que todos se encuentren dentro de un mismo contenedor.

Otros controles comunes

- **NotifyIcon**: es un control no visible en nuestra aplicación. Permite generar un icono en la barra de tareas de Windows con funciones u opciones que nosotros establezcamos.

También soporta menú contextual.

- **PictureBox**: permite mostrar una imagen seleccionada desde nuestro disco rígido, como de Internet o algún recurso incluido en la aplicación que contenga una colección de imágenes. Soporta los formatos más comunes de imágenes (**GIF, PNG, JPG, TIF, BMP, ICO**) y también permite usar imágenes con transparencias.
- **ProgressBar**: muestra una **barra de progreso**, generalmente combinada con la ejecución de algún proceso complejo en nuestro sistema. Sus propiedades **Minimun** y **Maximun** establecen los parámetros mínimos y máximos de la barra, mientras que la propiedad **Value** setea el porcentaje de progreso.
- **ToolTip**: este **control no visible** permite establecer una etiqueta con un **texto descriptivo** sobre los controles de la aplicación, cuando posicionamos el puntero del mouse sobre alguno de ellos.
- **ListView**: permite mostrar en forma de lista una jerarquía de elementos. Estos pueden ser una colección de objetos, **registros de una base de datos, archivos/carpetas** del disco o **unidades** de la computadora.
- **TreeView**: muestra una colección de objetos en forma de **árbol**, partiendo de una raíz, y conteniendo **nodos** y **subnodos** según nuestra necesidad. El ejemplo más común donde encontramos el uso de **TreeView** y **ListView** es en **Windows Explorer** y en el panel de control, entre otros.

Controles contenedores

Los **controles contenedores** permiten que nuestra aplicación incluya varios controles de cualquier tipo dentro de sí misma, y así organizarlos mejor en la pantalla. Otras ventajas de estos controles es poder establecer una propiedad **visible** de manera grupal, como una propiedad **Enabled** también de manera grupal, sin necesidad de tener que cambiar dichas propiedades por cada uno de los controles.

- **GroupBox**: permite sectorizar información común dentro de una ventana. Es útil para agrupar controles **RadioButton** y **CheckBox**.

Panel: posee bordes transparentes, con lo cual podemos distribuir controles en un Form a través de sus diferentes regiones. Cuenta con más opciones que **GroupBox**, como **Dock** y **Anchor**, que nos ayudan a posicionar fácilmente el resto de los controles contenidos, y admite **scrollbars** para mostrar parte de su contenido en pantalla.

- **TabControl**: agrupa varios controles, que pueden organizarse a través de solapas (**TabPage**s). Cada una puede configurarse de manera independiente. Es muy útil para organizar mucha información por categorías y cuando un **Form** de pequeñas dimensiones debe contener muchos controles.

Controles de menús y barras de herramientas

ToolStripContainer: es un objeto contenedor que divide nuestro **Form** en **tres paneles horizontales**. El superior agrupa la **barra de herramientas** de la aplicación, como también la **barra de menús** si es que la hay. El panel del medio permite establecer todo el **contenido de la ventana** con el que el usuario suele interactuar.

El inferior se utiliza para agrupar una **barra de tareas**.

- **MenuStrip**: con él podemos establecer los menús que tendrá nuestra aplicación.
 - **ContextMenuStrip**: con la misma funcionalidad que **MenuStrip**, permite establecer menús en nuestros desarrollos, que solo serán visualizados a través del botón derecho del mouse, previo código que lo invoque.
 - **ToolStrip**: permite crear una barra de herramientas en la aplicación, que podrá contener varios botones, personalizables al máximo en cuanto a estética y funciones, al igual que las barras de cualquier aplicación.
- Controles de acceso a datos

Los **controles de acceso** a datos nos permiten establecer conexión con una base de datos para recuperar los registros de las **tablas** y trabajar con ellos, así como también para mostrar **vistas de datos** y realizar

operaciones como **Alta, Baja y Modificación** de registros, o llamadas a **Stored Procedures** que permitan efectuar otras operaciones.

Convenciones para nombrar los controles

CONTROL	NOMBRE TENTATIVO	EJEMPLO
Button	btnNombreDeAccion	btnNuevoDocumento
Label	lblNombre	lblHipervinculo
TextBox	txtNombre	txtApellido
CheckBox	chkControl	chkLenguajeVBNET
RadioButton	rdbControl	rdbExperienciaBasica
ListView	lvControl	lvArchivos
MonthCalendar	mCalNombre	mCalAgenda
DataGrid	dgControl	dgClientes
PictureBox	picImagen	picFotoPerfilDerecho

Try Catch Finally

Try: código que se intenta ejecutar en un programa

Catch: código que se ejecuta en caso de que el código a tratar de ejecutar tenga algún error:

Finally: Finally puede utilizarse claramente para volver a vaciar variables del tipo Global o restablecer a Nothing determinados objetos para que no ocupen espacio en memoria, sobre todo, cuando trabajamos con archivos o conexiones de bases de datos.

Conexiones a base de datos:

DbCommand Clase

Representa una instrucción SQL o un procedimiento almacenado que se van a ejecutar en un origen de datos. Proporciona una clase base para las clases específicas de la base de datos que representan comandos.

CommandType Obtiene o establece cómo se interpreta la CommandText propiedad.

CommandText Obtiene o establece el comando de texto para ejecutar con el origen de datos.

Connection Obtiene o establece el objeto DbConnection utilizado por este DbCommand.

Parameters Obtiene la colección de objetos DbParameter. Para más información sobre los parámetros, vea Configurar parámetros y tipos de datos de parámetros.

Métodos

CreateDbParameter() Crea una nueva instancia de un objeto DbParameter.

ExecuteNonQuery() Ejecuta el comando en su objeto de conexión y devuelve el número de filas afectadas.

ExecuteReader() Ejecuta el comando en su conexión y devuelve un DbDataReader objeto que se puede usar para tener acceso a los resultados.

DbConnection Clase

Define el comportamiento básico de las conexiones de base de datos y proporciona una clase base para las conexiones específicas de base de datos.