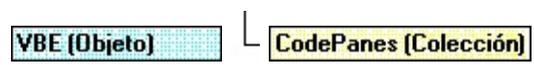


CodePanes (Colección)



Contiene los [paneles de código](#) activos en el objeto **VBE**.

Comentarios

Utilice la colección **CodePanels** para tener acceso a los paneles de código abiertos en un [proyecto](#).

Puede utilizar la propiedad **Count** para devolver el número de paneles de código activos en una [colección](#).

CommandBars (Colección)

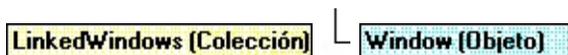


Contiene todas las barras de comandos de un [proyecto](#), incluyendo las barras de comandos que admiten los menús de método abreviado.

Comentarios

Utilice la colección **CommandBars** para permitir a los [complementos](#) agregar barras de comandos y [controles](#) o agregar controles a las barras de comandos existentes e incorporadas.

LinkedWindows (Colección)



Contiene todas las ventanas vinculadas en un [marco de ventana vinculada](#).

Comentarios

Utilice la colección **LinkedWindows** para modificar el estado [anclado](#) y [vinculado](#) de las ventanas en el [entorno de programación](#).

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución si se ejecutan en Macintosh.

La propiedad **LinkedWindowFrame** del objeto **Window** devuelve un objeto **Window** que tiene una colección **LinkedWindows** válida.

Los marcos de ventanas vinculados contienen todas las ventanas que se pueden vincular o anclar. Incluye todas las ventanas excepto las ventanas de código, [diseñadores](#), la ventana [Examinador de objetos](#), y la ventana Buscar y sustituir.

Si todos los paneles de un marco de ventana vinculado se mueven a otra ventana, se destruye el marco de la ventana vinculada sin paneles. Sin embargo, si se

eliminan todos los paneles de la ventana principal, éste no se destruye.

Utilice la propiedad **Visible** para comprobar o establecer la visibilidad de una ventana.

Puede utilizar el método **Add** para agregar una ventana a la [colección](#) de ventanas vinculadas actualmente. Una ventana que sea un panel en un marco de ventana vinculado se puede agregar a otro marco de ventana vinculado. Utilice el método **Remove** para eliminar una ventana de la colección de ventanas vinculadas actualmente; esto se produce en la ventana al ser desvinculada o desanclada.

La colección **LinkedWindows** se usa para anclar y desanclar ventanas del marco de la ventana principal.

Properties (Colección, Modelo de objeto de complementos de VBA)



Representa las [propiedades](#) de un objeto.

Comentarios

Utilice la colección **Properties** para tener acceso a las propiedades mostradas en la [ventana Propiedades](#). Para cada propiedad listada en la ventana Propiedades, hay un objeto en la colección **Properties**.

References (Colección)

References (Colección) L Reference (Objeto)

Representa el conjunto de referencias en el [proyecto](#).

Comentarios

Utilice la colección **References** para agregar o eliminar referencias. La colección **References** es igual que el conjunto de referencias seleccionado en el cuadro de diálogo **References**.

VBComponents (Colección)



Representa los componentes contenidos en un [proyecto](#).

Comentarios

Utilice la colección **VBComponents** para tener acceso, agregar o quitar componentes en un proyecto. Un componente puede ser un [formulario](#), [módulo](#) o [clase](#). La colección **VBComponents** es una [colección](#) estándar que se puede utilizar en un bloque **For Each**.

Puede utilizar la propiedad **Parent** para volver al proyecto en el que está la colección **VBComponents**.

En Visual Basic para Aplicaciones, puede utilizar el método **Import** para agregar un componente a un proyecto desde un archivo.

VBProjects (Colección)

VBE (Objeto) L **VB**Projects (Colección)

Representa todos los [proyectos](#) que están abiertos en el [entorno de programación](#).

Comentarios

Utilice la colección **VBProjects** para tener acceso a proyectos especificados en una instancia del entorno de desarrollo. **VBProjects** es una colección estándar

que puede usarse en un bloque **For Each**.

Windows (Colección)



Contiene todas las ventanas permanentes o abiertas.

Comentarios

Utilice la colección **Windows** para tener acceso a los objetos **Window**.

La colección **Windows** tiene un conjunto de ventanas fijo que siempre están disponibles en la [colección](#), como son la [ventana Proyecto](#), la [ventana Propiedades](#) y un conjunto de ventanas que representan todas las ventanas de código abiertas y ventanas de [diseñadores](#). Al abrir una ventana de diseñador o de código se añade un nuevo miembro a la colección **Windows**. Al cerrar una ventana de diseño o un código se elimina un miembro de la colección **Windows**. Al cerrar una ventana de [entorno de programación](#) permanente, no se elimina el objeto correspondiente de esta colección, pero aparece en la ventana de manera no visible.

Click (Evento, modelo de objeto de complementos de VBA)

Se produce cuando se establece la [propiedad **OnAction**](#) de un control de barra de comandos correspondiente.

Sintaxis

Sub objeto_Click (ByVal ctrl As Object, ByRef handled As Boolean, ByRef canceldefault As Boolean)

La sintaxis del evento **Click** tiene estos [argumentos con nombre](#):

Parte	Descripción
<i>ctrl</i>	Requerido; tipo de datos Object . Especifica el objeto origen del evento Click .
<i>handled</i>	Requerido; tipo de datos Boolean . Si es True , otros complementos deben controlar el evento. Si es False , no se ha controlado la acción del elemento de la barra de comandos.
<i>canceldefault</i>	Requerido; Boolean . Si es True , se ejecuta el comportamiento predeterminado a menos que se cancele por la siguiente macro automática. Si es False , no se ejecuta el comportamiento predeterminado a menos que lo restaure una macro automática.

Comentarios

El evento **Click** es específico del objeto **CommandBarEvents**. Utilice una [variable](#) declarada mediante la palabra clave **WithEvents** para recibir el evento **Click** para un control **CommandBar**. Esta variable se debe establecer al valor devuelto por la propiedad **CommandBarEvents** del objeto **Events**. La propiedad **CommandBarEvents** obtiene el control **CommandBar** como un [argumento](#).

Cuando se hace clic en el control **CommandBar** (para la variable que declaró mediante la palabra clave **WithEvents**), se ejecuta el código.

ItemAdded (Evento, modelo de objeto de complementos de VBA)

Se produce después de agregar una referencia.

Sintaxis

Sub objeto_ItemAdded(ByVal elemento As Reference)

El [argumento](#) requerido *elemento* especifica el elemento que se agregó.

Comentarios

El evento ***ItemAdded*** se produce cuando se agrega un objeto ***Reference*** a la colección ***References***.

ItemRemoved (Evento, modelo de objeto de complementos de VBA)

Se produce después de quitar una referencia de un [proyecto](#).

Sintaxis

Sub objeto_ItemRemoved(ByVal elemento As Reference)

El [argumento](#) requerido *elemento* especifica el objeto ***Reference*** que se quitó.

Add (Método, modelo de objeto de complementos de VBA)

Agrega un objeto a una [colección](#).

Sintaxis

objeto.**Add**(*componente*)

La sintaxis del método **Add** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>componente</i>	Requerido. Para la colección LinkedWindows , un objeto. Para la colección VBComponents , una constante enumerada que representa un módulo de clase , un formulario o un módulo estándar . Para la colección VBProjects , una constante enumerada representa un tipo de proyecto.

Puede utilizar una de las siguientes constantes para el argumento *componente*:

Constante	Descripción
vbext_ct_ClassModule	Agrega un módulo de clase a la colección.
vbext_ct_MSForm	Agrega un formulario a la colección.
vbext_ct_StdModule	Agrega un módulo estándar a la colección.
vbext_pt_StandAlone	Agrega un proyecto individual a la colección.

Comentarios

Para la colección **LinkedWindows**, el método **Add** agrega una ventana a la colección de las [ventanas vinculadas](#) actualmente.

Nota Puede agregar una ventana que es un panel en un [marco de ventana vinculada](#) a otro marco de ventana vinculada; la ventana simplemente se mueve de un panel a otro. Si el marco de ventana vinculada del que se mueve la ventana no contiene ningún panel, se destruye.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventana vinculados y ventanas acopladas se incluyen en Macintosh por compatibilidad con código escrito en Windows. Sin embargo, estos elementos de lenguaje generan errores de tiempo de ejecución al ejecutarse en Macintosh.

Para la colección **VBComponents**, el método **Add** crea un nuevo componente estándar y lo agrega al [proyecto](#).

Para la colección **VBComponents**, el método **Add** devuelve un objeto **VBComponent**. Para la colección **LinkedWindows**, el método **Add** devuelve **Nothing**.

Para la colección **VBProjects**, el método **Add** devuelve un objeto **VBProject** y agrega un proyecto a la colección **VBProjects**.

AddFromFile (Método)

Para la colección **References**, agrega una referencia a un [proyecto](#) de un archivo.
Para el objeto **CodeModule**, agrega el contenido de un archivo a un [módulo](#).

Sintaxis

objeto.**AddFromFile**(*nombreamodulo*)

La sintaxis del método **AddFromFile** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>nombreamodulo</i>	Requerido. Una expresión de cadena que especifica el nombre del archivo que desea agregar al proyecto o al módulo. Si el archivo no se encuentra y no se especifica una ruta de acceso, se busca en los directorios especificados en la función Windows OpenFile .

Comentarios

Para el objeto **CodeModule**, el método **AddFromFile** inserta el contenido del archivo comenzando en la línea anterior al primer [procedimiento](#) en el [módulo de código](#). Si el módulo no contiene procedimientos, **AddFromFile** coloca el contenido del archivo al final del módulo.

AddFromGuid (Método)

Agrega una referencia a la colección **References** mediante el identificador único global (GUID) de la referencia.

Sintaxis

objeto.**AddFromGuid**(*guid*, *principal*, *secundaria*) **As Reference**

La sintaxis del método **AddFromGuid** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>guid</i>	Requerido. Una expresión de cadena que representa el GUID de la referencia.
<i>principal</i>	Requerido. Un valor de tipo Long que especifica el número de versión principal de la referencia.
<i>secundaria</i>	Requerido. Un valor de tipo Long que especifica el número de versión secundario de la referencia.

Comentarios

El método **AddFromGuid** busca en el [registro](#) para encontrar la referencia que desea agregar. El GUID puede ser una [biblioteca de tipos](#), un [control](#), un identificador de clase, etc.

AddFromString (Método)

Agrega texto a un [módulo](#).

Sintaxis

objeto.**AddFromString**

El marcador de posición *objeto* es una [expresión de objeto](#) que da como resultado un objeto incluido en la lista Se aplica a.

Comentarios

El método **AddFromString** inserta el texto comenzando en la línea anterior al primer [procedimiento](#) del módulo. Si el módulo no contiene procedimientos, **AddFromString** coloca el texto al final del módulo.

Close (Método, modelo de objeto de complementos de VBA)

Cierra y destruye una ventana.

Sintaxis

objeto.**Close**

El marcador de posición *objeto* es una [expresión de objeto](#) que da como resultado un objeto incluido en la lista Se aplica a.

Comentarios

Los siguientes tipos de ventanas responden al método **Close** de formas diferentes:

Para una ventana que es un [panel de código](#), **Close** destruye el panel de código.

Para una ventana que es un [diseñador](#), **Close** destruye el diseñador contenido.

Para las ventanas que siempre están disponibles en el menú **Ver**, **Close** oculta la ventana.

CreateEventProc (Método)

Crea un [procedimiento](#) de evento.

Sintaxis

objeto.CreateEventProc(*nombreevento*, *nombrequeto*) As Long

La sintaxis del método **CreateEventProc** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>nombreevento</i>	Requerido. Una expresión de cadena que especifica el nombre del evento que desea agregar al módulo .
<i>nombrequeto</i>	Requerido. Una expresión de cadena que especifica el nombre del objeto que es el origen del evento.

Comentarios

Utilice el método **CreateEventProc** para crear un procedimiento de evento. Por ejemplo, para crear un procedimiento de evento para el evento Click de un control **Command Button** llamado `command1` debe utilizar el código siguiente, donde `CM` representa un objeto de tipo **CodeModule**:

```
UbicaciónTexto = CM.CreateEventProc("Click", "C
```

El método **CreateEventProc** devuelve la línea en la que comienza el cuerpo del procedimiento de evento. **CreateEventProc** falla si los [argumentos](#) hacen referencia a un evento que no existe.

DeleteLines (Método)

Elimina una única línea o un intervalo de líneas especificado.

Sintaxis

objeto.DeleteLines (*líneainicio* [, *cuenta*])

La sintaxis del método **DeleteLines** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>líneainicio</i>	Requerido. Un valor de tipo Long que especifica la primera línea que desea eliminar.
<i>cuenta</i>	Opcional. Un valor de tipo Long que especifica el número de líneas que desea eliminar.

Comentarios

Si no especifica cuántas líneas desea eliminar, **DeleteLines** elimina una línea.

Export

(Método, modelo de objeto de complemento de VBA)

Guarda un componente como un archivo o archivos separados.

Sintaxis

objeto.**Export**(*nombreamarchivo*)

La sintaxis del método **Export** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>nombreamarchivo</i>	Requerido. Un valor de tipo String que especifica el nombre del archivo al que desea exportar el componente.

Comentarios

Cuando utiliza el método **Export** para guardar un componente como un archivo o archivos separados, utilice un nombre de archivo que no exista; de lo contrario se producirá un error.

Find (Método, modelo de objeto de complemento de VBA)

Busca en el [módulo](#) activo una cadena especificada.

Sintaxis

objeto.**Find**(*destino*, *líneainicio*, *colinicio*, *líneafin*, *colfin* [, *palabraCompleta*] [, *coincidirmaymin*] [, *modelobúsqueda*]) **As Boolean**

La sintaxis del método **Find** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>destino</i>	Requerido. Un valor de tipo String que contiene el texto o el modelo que desea encontrar.
<i>líneainicio</i>	Requerido. Un valor de tipo Long que especifica la línea en la que desea comenzar la búsqueda; si se encuentra una será el valor de la línea de coincidencia. La primera línea es la número uno.
<i>colinicio</i>	Requerido. Un valor de tipo Long que especifica la columna en la que desea comenzar la búsqueda; si se encuentra una será el valor de la columna de coincidencia. La primera columna es la número uno.
<i>líneafin</i>	Requerido. Un valor de tipo Long que especifica la última línea de la coincidencia si se encuentra una. La última línea puede designarse como -1.
<i>colfin</i>	Requerido. Un valor de tipo Long que especifica la última columna de la coincidencia si se encuentra una. La última columna puede designarse como -1.
<i>palabraCompleta</i>	Opcional. Un valor de tipo Boolean que especifica si sólo coinciden palabras completas. Si es True , sólo coinciden palabras completas. El valor predeterminado es False .
<i>coincidirmaymin</i>	Opcional. Un valor de tipo Boolean que especifica si coinciden mayúsculas y minúsculas. Si es True , la búsqueda distingue mayúsculas de minúsculas. El valor predeterminado es False .
<i>modelobúsqueda</i>	Opcional. Un valor de tipo Boolean que especifica si la cadena de destino es un modelo de expresión normal. Si True , es la cadena de destino es un modelo de expresión normal. El

valor predeterminado es **False**.

Comentarios

Find devuelve **True** si se encuentra una coincidencia y **False** si no se encuentra ninguna.

Los [argumentos](#) *coincidirmaymin* y *modelobúsqueda* se excluyen mutuamente; si se transfieren ambos argumentos como **True**, se produce un error.

El método **Find** no afecta al contenido del cuadro de diálogo **Buscar**.

El intervalo de líneas y columnas especificado es inclusivo, de forma que una búsqueda puede encontrar el modelo en la última línea especificada si *colfin* se proporciona como -1 o la longitud de la línea.

GetSelection (Método)

Devuelve la selección en un [panel de código](#).

Sintaxis

objeto.**GetSelection**(*líneainicio*, *colinicio*, *líneafin*, *colfin*)

La sintaxis del método **GetSelection** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>líneainicio</i>	Requerido. Un valor de tipo Long que devuelve un valor que especifica la primera línea de la selección en el panel de código.
<i>colinicio</i>	Requerido. Un valor de tipo Long que devuelve un valor que especifica la primera columna de la selección en el panel de código.
<i>líneafin</i>	Requerido. Un valor de tipo Long que devuelve un valor que especifica la última línea de la selección en el panel de código.
<i>colfin</i>	Requerido. Un valor de tipo Long que devuelve un valor que especifica la última columna de la selección en el panel de código.

Comentarios

Cuando utilice el método **GetSelection**, la información se devuelve en [argumentos](#) de salida. Como resultado, deben transferirse [variables](#) porque éstas se modificarán para contener la información cuando se devuelva.

Import (Método, modelo de objeto de complemento de VBA)

Agrega un componente a un [proyecto](#) de un archivo; devuelve el componente agregado recientemente.

Sintaxis

objeto.**Import**(*nombreamodulo*) As **VBComponent**

La sintaxis del método **Import** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>nombreamodulo</i>	Requerido. Un valor de tipo String que especifica la ruta de acceso y el nombre del archivo del que desea importar el componente.

Comentarios

Puede utilizar el método **Import** para agregar un componente, [formulario](#), [módulo](#), [clase](#), u otros al proyecto.

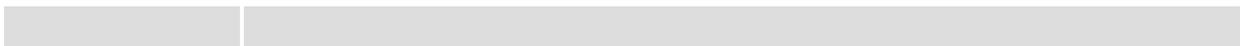
InsertLines (Método)

Inserta una línea o líneas de código en una ubicación especificada en un bloque de código.

Sintaxis

objeto.**InsertLines**(línea, código)

La sintaxis del método **InsertLines** consta de las siguientes partes:



Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>línea</i>	Requerido. Un valor de tipo Long que especifica la ubicación en la que desea insertar el código.
<i>código</i>	Requerido. Un valor de tipo String que contiene el código que desea insertar.

Comentarios

Si el texto que inserta mediante el método **InsertLines** está delimitado por retorno de carro-avance de línea, se insertarán como líneas consecutivas.

Item (Método, modelo de objeto de complemento de VBA)

Devuelve el miembro indexado de una [colección](#).

Sintaxis

objeto.**Item**(*índice*)

La sintaxis del método **Item** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>índice</i>	Requerido. Una expresión que especifica la posición de un miembro de la colección. Si es una expresión numérica , <i>índice</i> debe ser un número entre 1 y el valor de la propiedad Count de la colección. Si es una expresión de cadena , <i>índice</i> debe corresponder al argumento <i>key</i> especificado cuando se agrega el miembro a la colección.

La siguiente tabla enumera las colecciones y sus correspondientes argumentos *key* para utilizarlos con el método **Item**. La cadena que transfiere al método **Item** debe coincidir con el argumento *key* de la colección.

Colección	Argumento Key
Windows	Valor de la propiedad Caption
LinkedWindows	Valor de la propiedad Caption
CodePanels	Ninguna cadena única está asociada con esta colección.
VBProjects	Valor de la propiedad Name
VBComponents	Valor de la propiedad Name
References	Valor de la propiedad Name
Properties	Valor de la propiedad Name

Comentarios

El argumento *índice* puede ser un valor numérico o una cadena que contiene el título del objeto.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventana vinculados y ventanas acopladas se incluyen en

Macintosh por compatibilidad con código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores de tiempo de ejecución al ejecutarse en Macintosh.

Remove (Método, modelo de objeto de complemento de VBA)

Quita un elemento de una [colección](#).

Sintaxis

objeto.**Remove**(*componente*)

La sintaxis de **Remove** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Requerido. Una expresión de objeto que da como resultado un objeto incluido en la lista Se aplica a.
<i>componente</i>	Requerido. Para la colección LinkedWindows , un objeto. Para la colección References , una referencia a una biblioteca de tipos o un proyecto . Para la colección VBComponents , una constante enumerada que representa un módulo de clase , un formulario o un módulo estándar . Para la colección VBProjects , un proyecto individual.

Comentarios

Cuando se utiliza en la colección **LinkedWindows**, el método **Remove** quita una ventana de la colección de [ventanas vinculadas](#) actualmente. La ventana que se quita se convierte en una ventana flotante que tiene su propio [marco de ventana vinculado](#). El método **Remove** sólo puede usarse en un proyecto único.

Genera un error en tiempo de ejecución si se intenta usar en un proyecto host.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventana vinculados y ventanas acopladas se incluyen en Macintosh por compatibilidad con código escrito en Windows. Sin embargo, estos elementos de lenguaje generan errores de tiempo de ejecución al ejecutarse en Macintosh.

ReplaceLine (Método)

Reemplaza una línea de código existente con una línea de código especificada.

Sintaxis

objeto.**ReplaceLine**(línea, código)

La sintaxis de **ReplaceLine** consta de las siguientes partes:

Parte	Descripción
--------------	--------------------

- objeto* Requerido. Una [expresión de objeto](#) que se evalúa como un objeto de la lista Se aplica a.
- línea* Requerido. Un valor de tipo [Long](#) que especifica la ubicación de la línea que desea reemplazar.
- código* Requerido. Un valor de tipo [String](#) que contiene el código que desea insertar.

SetFocus (Método)

Mueve el [enfoque](#) a la ventana especificada.

Sintaxis

objeto.**SetFocus**

El marcador de posición *objeto* es una [expresión de objeto](#) que se evalúa como un objeto de la lista Se aplica a.

Comentarios

Utilice el método **SetFocus** en ventanas que ya son visibles.

SetSelection (Método)

Establece la selección en el [panel de código](#).

Sintaxis

objeto.**SetSelection**(*líneainicio*, *colinicio*, *líneafin*, *colfin*)

La sintaxis de **SetSelection** consta de las siguientes partes:

Parte	Descripción
--------------	--------------------

<i>objeto</i>	Requerido. Una expresión de objeto que se evalúa como un objeto de la lista Se aplica a.
<i>líneainicio</i>	Requerido. Un valor de tipo Long que especifica la primera línea de la selección.
<i>colinicio</i>	Requerido. Un valor de tipo Long que especifica la primera columna de la selección.
<i>líneafin</i>	Requerido. Un valor de tipo Long que especifica la última línea de la selección.
<i>colfin</i>	Requerido. Un valor de tipo Long que especifica la última columna de la selección.

CodeModule (Objeto)

CodeModule (Objeto)

Representa el código asociado a un componente, como un [formulario](#), una [clase](#) o un [documento](#).

Comentarios

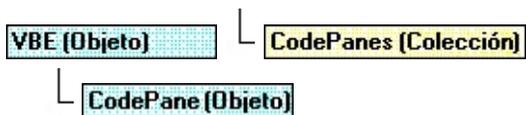
Utilice el objeto **CodeModule** para cambiar (agregar, eliminar o modificar) el código asociado a un componente.

Cada componente está asociado a un objeto **CodeModule**. Sin embargo, un objeto **CodeModule** se puede asociar con múltiples [paneles de código](#).

Los métodos asociados con el objeto **CodeModule** le permiten manipular y devolver información acerca del texto de código línea a línea. Por ejemplo, puede utilizar el método **AddFromString** para agregar el texto al [módulo](#). **AddFromString** sitúa el texto delante del primer [procedimiento](#) en el módulo o al final del módulo si no existen procedimientos.

Utilice la propiedad **Parent** para devolver el objeto **VBComponent** asociado a un [módulo de código](#).

CodePane (Objeto)



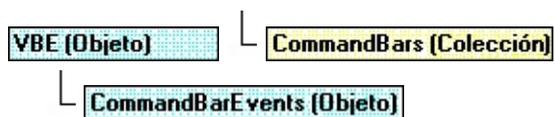
Representa un [panel de código](#).

Comentarios

Utilice el objeto **CodePane** para manipular la posición de un texto visible o de selección de texto mostrado en el panel de código.

Puede utilizar el método **Show** para hacer visible el panel de código que especifique. Utilice el método **SetSelection** para establecer la selección en un panel de código y el método **GetSelection** para devolver la ubicación de la selección en el panel de código.

CommandBarEvents (Objeto)



Devuelto por la propiedad **CommandBarEvents**. El objeto **CommandBarEvents** activa un evento cuando se hace clic en un [control](#) de la barra de comandos.

Comentarios

El objeto **CommandBarEvents** se devuelve mediante la propiedad **CommandBarEvents** del objeto **Events**. El objeto devuelto tiene un evento en su interfaz, el evento Click. Puede controlar este evento mediante la declaración del objeto **WithEvents**.

Events (Objeto)

Proporciona [propiedades](#) que permiten a los [complementos](#) conectar todos los eventos en Visual Basic para aplicaciones.

Comentarios

El objeto **Events** proporciona propiedades que devuelven [objetos origen de eventos](#). Utilice las propiedades para devolver objetos de origen de eventos que lo informarán de los cambios en el entorno de trabajo de Visual Basic para aplicaciones.

Las propiedades del objeto **Events** devuelve objetos del mismo tipo que el nombre de la propiedad. Por ejemplo, la propiedad **CommandBarEvents** devuelve el objeto **CommandBarEvents**.

Property (Objeto)



Representa las [propiedades](#) de un objeto que son visibles en la [ventana Propiedades](#) para cualquier componente dado.

Comentarios

Utilice la propiedad **Value** del objeto **Property** para devolver o establecer el valor de una propiedad de un componente.

Como mínimo, todos los componentes tienen una propiedad **Name**. Utilice la propiedad **Value** del objeto **Property** para devolver o establecer el valor de la propiedad. La propiedad **Value** devuelve un [tipo de datos Variant](#) del tipo apropiado. Si el valor devuelto es un objeto, la propiedad **Value** devuelve la colección **Properties** que contiene objetos **Property** que representa las propiedades individuales del objeto. Puede tener acceso a cada uno de los objetos **Property** usando el método **Item** de la colección **Properties** devuelta.

Si el valor devuelto por el objeto **Property** es un objeto, puede utilizar la propiedad **Object** para establecer el objeto **Property** a un nuevo objeto.

Reference (Objeto)

References (Colección) | **Reference (Objeto)**

Representa una referencia a una [biblioteca de tipos](#) o a un [proyecto](#).

Comentarios

Utilice el objeto **Reference** para comprobar si una referencia es todavía válida.

La propiedad **IsBroken** devuelve **True** si la referencia ya no apunta a una

referencia válida. La propiedad **BuiltIn** devuelve **True** si la referencia es una referencia predeterminada que no se puede mover o eliminar. Utilice la propiedad **Name** para determinar si la referencia que quiere agregar o eliminar es la correcta.

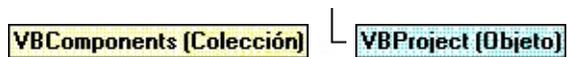
ReferencesEvents (Objeto)

Devuelto por la propiedad **ReferencesEvents**.

Comentarios

El objeto **ReferencesEvents** es el origen de los eventos que ocurren cuando se agrega o elimina una referencia de un [proyecto](#). El evento ItemAdded se activa después de que una referencia se agrega a un proyectos. El evento ItemRemoved se activa después de que una referencia se elimina de un proyecto.

VBComponent (Objeto)



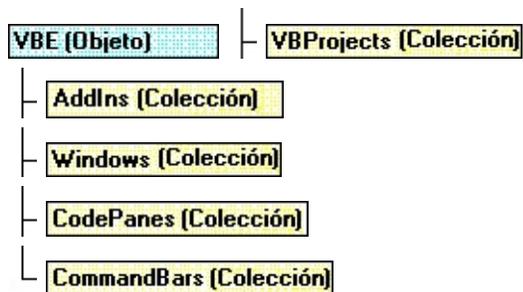
Representa un componente, como un [módulo de clase](#) o un [módulo estándar](#), contenido en un [proyecto](#).

Comentarios

Utilice el objeto **VBComponent** para tener acceso al [módulo de código](#) asociado con un componente o para cambiar los valores de las propiedades de un componente.

Puede utilizar la propiedad **Type** para saber a qué tipo de componente se refiere el objeto **VBComponent**. Utilice la propiedad **Collection** para saber en qué [colección](#) está el componente.

VBE (Objeto)



El objeto raíz que contiene todos los demás [objetos](#) y [colecciones](#) representados en Visual Basic para aplicaciones.

Comentarios

Puede utilizar colecciones para tener acceso a los objetos contenidos en el objeto **VBE**:

Utilice la colección **VBProjects** para tener acceso a la colección de [proyectos](#).

Utilice la colección **AddIns** para tener acceso a la colección de complementos

Utilice la colección **Windows** para tener acceso a la colección de ventanas.

Utilice la colección **CodePanels** para tener acceso a la colección de [paneles de código](#).

Utilice la colección **CommandBars** para tener acceso a la colección de

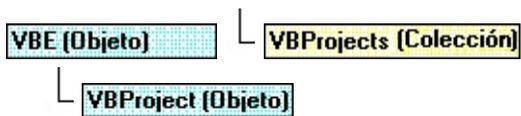
barras de comandos.

Utilice la colección **Events** para tener acceso a las propiedades que permitan [complementos](#) para conectar todos los eventos en Visual Basic para aplicaciones. Las propiedades del objeto **Events** retornan objetos del mismo tipo, como el nombre de la propiedad. Por ejemplo, la propiedad **CommandBarEvents** retorna el objeto **CommandBarEvents**.

Puede utilizar la propiedad **SelectedVBComponent** para devolver el componente activo. El componente activo es el componente al que se sigue en la [ventana Proyecto](#). Si el elemento seleccionado en la ventana Proyecto no es un componente, **SelectedVBComponent** devuelve **Nothing**.

Nota Todos los objetos de este modelo de objeto tienen una propiedad **VBE** que apunta al objeto **VBE**.

VBProject (Objeto)



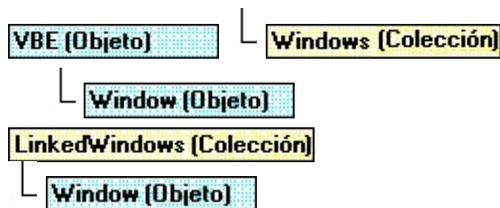
Representa un [proyecto](#).

Comentarios

Utilice el objeto **VBProject** para establecer [propiedades](#) para el proyecto, para

tener acceso a la colección **VBComponents** y para tener acceso a la colección **References**.

Window (Objeto)



Representa una ventana en el [entorno de programación](#).

Comentarios

Utilice el objeto **Window** para mostrar, ocultar o ubicar ventanas.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventana vinculados y ventanas acopladas se incluyen en Macintosh por compatibilidad con código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores de tiempo de ejecución al ejecutarse en Macintosh.

Puede utilizar el método **Close** para cerrar una ventana en la colección **Windows**. El método **Close** afecta a diferentes tipos de ventanas como se muestra a continuación:

Ventana	Resultado de utilizar el método Close
Ventana de código	Elimina la ventana de la colección Windows .
Diseñador	Elimina la ventana de la colección Windows .
Objetos Window del tipo marco de ventana vinculada	Las ventanas se hacen ventanas separadas no vinculadas.

Nota Al utilizar el método **Close** con los diseñadores y las ventanas de código se cierra la ventana. Establecer la propiedad **Visible** a **False** oculta la ventana pero no la cierra. Usar el método **Close** con las ventanas del entorno de desarrollo, tales como la [ventana Proyecto](#) o la [ventana Propiedades](#), es lo mismo que establecer la propiedad **Visible** a **False**.

Puede utilizar el método **SetFocus** para mover el [enfoco](#) a una ventana.

Puede utilizar la propiedad **Visible** para devolver o establecer la visibilidad de una ventana.

Para averiguar con qué tipo de ventana está trabajando, puede utilizar la propiedad **Type**. Si tiene más de una ventana de un tipo, por ejemplo, varios diseñadores, puede utilizar la propiedad **Caption** para determinar la ventana con la que está trabajando. También puede encontrar la ventana con la que quiere trabajar usando la propiedad **DesignerWindow** del objeto **VBComponent** o de la propiedad **Window** del objeto **CodePane**.

ActiveCodePane (Propiedad)

Devuelve el objeto **CodePane** activo o último objeto **CodePane** activo o establece el objeto **CodePane** activo. Lectura/escritura.

Comentarios

Puede establecer la propiedad **ActiveCodePane** a cualquier objeto **CodePane** válido, como se muestra en el siguiente ejemplo:

```
Set MiAppli.VBE.ActiveCodePane = MiAppli.VBE.Code
```

El ejemplo anterior establece el primer [panel de código](#) en una [colección](#) de paneles de código como el panel de código activo. También puede activar un panel de código con el método **Set**.

ActiveVBProject (Propiedad)

Devuelve el [proyecto](#) activo en la [ventana Proyecto](#). Esta propiedad es de sólo lectura.

Comentarios

La propiedad **ActiveVBProject** devuelve el proyecto seleccionado en la ventana Proyecto o el proyecto en el que está seleccionado el componente. En este último caso el proyecto no está necesariamente seleccionado. Aunque el proyecto esté o no seleccionado explícitamente siempre hay un proyecto activo.

ActiveWindow (Propiedad)

Devuelve la ventana activa en el [entorno de programación](#). Esta propiedad es de sólo lectura.

Comentarios

Cuando se abre más de una ventana en el entorno de desarrollo, la propiedad **ActiveWindow** contiene la ventana con el [enfoco](#). Si la ventana principal tiene el enfoque, **ActiveWindow** devuelve **Nothing**.

BuiltIn (Propiedad)

Devuelve un valor [Booleano](#) que indica si la referencia es o no una referencia predeterminada que no se puede eliminar. Sólo lectura.

Valores devueltos

La propiedad **BuiltIn** devuelve estos valores:

Valor	Descripción
True	La referencia es un valor predeterminado que no se puede eliminar.
False	La referencia no es un valor predeterminado, se puede eliminar.

Caption (Propiedad)

Devuelve un [tipo de datos String](#) que contiene el título de la ventana activa. Esta propiedad es de sólo lectura.

Comentarios

El título de la ventana activa es el texto mostrado en la barra de título de la ventana.

CodeModule (Propiedad)

Devuelve un objeto que representa el código asociado a un componente. Esta propiedad es de sólo lectura.

Comentarios

La propiedad **CodeModule** devuelve **Nothing** si el componente no tiene un [módulo de código](#) asociado a él.

Nota El objeto **CodePane** representa una ventana de código visible. Un componente dado puede tener varios objetos **CodePane**. El objeto **CodeModule** representa el código dentro de un componente. Un componente sólo puede tener un objeto **CodeModule**.

CodePane (Propiedad)

Devuelve un objeto **CodePane**. Esta propiedad es de sólo lectura.

Comentarios

Si un [panel de código](#) existe, pasa a ser el panel de código activo y la ventana que lo contiene pasa a ser la ventana activa. Si un panel de código no existe para el [módulo](#), la propiedad **CodePane** crea uno.

CodePanes (Propiedad)

Devuelve la [colección](#) de objetos **CodePane** activos. Sólo lectura.

CodePaneView (Propiedad)

Devuelve un valor que indica si el [panel de código](#) está en modo de ver procedimientos o ver módulo completo. Esta propiedad es de sólo lectura.

Valores devueltos

La propiedad **CodePaneView** devuelve estos valores:

Constante	Descripción
vbext_cv_ProcedureView	El panel de código especificado está en modo de ver procedimientos.
vbext_cv_FullModuleView	El proyecto especificado está en modo de ver módulo completo.

Collection (Propiedad)

Devuelve la [colección](#) que contiene el objeto con el que está trabajando. Esta propiedad es de sólo lectura.

Comentarios

La mayoría de los objetos de este modelo de objeto tienen una propiedad **Parent** o una propiedad **Collection** que apunta al objeto padre del objeto.

Utilice la propiedad **Collection** para tener acceso a las [propiedades](#), [métodos](#) y

[controles](#) de la colección a la que pertenece el objeto.

CommandBarEvents (Propiedad)

Devuelve el objeto **CommandBarEvents**. Esta propiedad es de sólo lectura.

Valores

El valor del [argumento](#) que se le pasa a la propiedad **CommandBarEvents** es:

Argumento	Descripción
<i>vbcontrol</i>	Debe ser un objeto de tipo CommandBarControl .

Comentarios

Utilice la propiedad **CommandBarEvents** para devolver un [objeto origen de evento](#) que activa un evento cuando se hace clic sobre un botón de la barra de comandos. El argumento pasado a la propiedad **CommandBarEvents** es el control barra de comandos para el que se activará el evento **Click**.

Count (Propiedad)

Devuelve un [tipo de datos Long](#) que contiene el número de elementos de una [colección](#). Esta propiedad es de sólo lectura.

CountOfDeclarationLines (Propiedad)

Devuelve un [tipo de datos Long](#) que contiene el número de líneas de código de la sección de declaraciones de un [módulo de código](#). Esta propiedad es de sólo lectura.

CountOfLines (Propiedad)

Devuelve un [tipo de datos Long](#) que contiene el número de líneas de código de un [módulo de código](#). Esta propiedad es de sólo lectura.

CountOfVisibleLines (Propiedad)

Devuelve un [tipo de datos Long](#) que contiene el número de líneas visibles de un [panel de código](#). Esta propiedad es de sólo lectura.

Description (Propiedad)

Devuelve o establece una [expresión de cadena](#) que contiene una cadena descriptiva asociada a un objeto. Para el objeto **VBProject** es de lectura y escritura; para el objeto **Reference** es de sólo lectura.

Comentarios

Para el objeto **VBProject**, la propiedad **Description** devuelve o establece una cadena descriptiva asociada al [proyecto](#) activo.

Para el objeto **Reference**, la propiedad **Description** devuelve el nombre descriptivo de la referencia.

Designer (Propiedad)

Devuelve el objeto que le permite tener acceso a las características de diseño de un componente.

Comentarios

Si el objeto tiene un [diseñador](#) abierto, la propiedad **Designer** devuelve el diseñador abierto, en otro caso se crea un nuevo diseñador. El diseñador es una característica de algunos objetos **VBComponent**. Por ejemplo, cuando cree algunos tipos de objetos **VBComponent**, se crea un diseñador junto con el

objeto. Un componente puede tener sólo un diseñador y siempre es el mismo. La propiedad **Designer** le permite tener acceso a objetos específicos de componentes. En algunos casos como en los [módulos estándar](#) y [módulos de clase](#), no se crean diseñadores puesto que estos tipos de objetos **VBComponent** no admiten diseñadores.

La propiedad **Designer** devuelve **Nothing** si el objeto **VBComponent** no tiene un diseñador.

FullPath (Propiedad)

Devuelve una [cadena](#) que contiene el nombre de la ruta y el archivo de la [biblioteca de tipo](#) referenciada. Sólo lectura.

GUID (Propiedad)

Devuelve un [tipo de datos String](#) que contiene la clase que identifica a un objeto.
Esta propiedad es de sólo lectura.

HasOpenDesigner (Propiedad)

Devuelve un valor de [tipo de datos Boolean](#) que indica si el objeto **VBComponent** tiene o no un [diseñador](#) abierto. Esta propiedad es de sólo lectura.

Valores devueltos

La propiedad **HasOpenDesigner** devuelve estos valores:

Valor	Descripción
True	El objeto VBComponent tiene una ventana Diseño abierta.
False	El objeto VBComponent no tiene una ventana Diseño abierta.

Height (Propiedad, modelo de objeto de complementos de VBA)

Devuelve o establece un [tipo de datos Single](#) que contiene el alto de la ventana medido en [twips](#). Esta propiedad es de lectura y escritura.

Comentarios

Cambiar el valor de la propiedad **Height** de una [ventana vinculada](#) o de una [ventana acoplada](#) no tiene un efecto mientras que la ventana permanece vinculada o acoplada.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

HelpContextID (Propiedad, modelo de objeto de complementos de VBA)

Devuelve o establece un [tipo de datos String](#) que contiene el ID de contexto de un tema en un archivo de Ayuda de Microsoft Windows. Esta propiedad es de lectura y escritura.

HelpFile (Propiedad, modelo de objeto de complementos de VBA)

Devuelve o establece un [tipo de datos String](#) que especifica el archivo de Ayuda de Microsoft Windows de un [proyecto](#). Esta propiedad es de lectura y escritura.

IndexedValue (Propiedad, modelo de objeto de complementos de VBA)

Devuelve o establece un valor de un miembro de una [propiedad](#), puede ser una lista indexada o una [matriz](#).

Comentarios

El valor que se puede establecer u obtener mediante la propiedad **IndexedValue** es una [expresión](#) que evalúa a un tipo aceptado por el objeto. Para una propiedad que sea una lista indexada o una matriz, debe utilizar la propiedad **IndexedValue** en lugar de la propiedad **Value**. Una lista indexada es una [expresión numérica](#)

que especifica la posición del índice.

IndexedValue acepta hasta 4 índices. El número de índices aceptados por **IndexedValue** es el valor devuelto por la propiedad **NumIndices**.

La propiedad **IndexedValue** se utiliza sólo si el valor de la propiedad **NumIndices** es mayor que cero. Los valores en listas indexadas se establecen o devuelven con un índice único.

IsBroken (Propiedad)

Devuelve un valor de [tipo de datos Boolean](#) indicando si el objeto **Reference** apunta a una referencia válida en el [registro](#). Esta propiedad es de solo lectura.

Valores devueltos

La propiedad **IsBroken** devuelve estos valores:

Valor	Descripción
True	El objeto Reference ya no apunta a una referencia válida en el registro.
False	El objeto Reference apunta a una referencia válida en el registro.

Left (Propiedad)

Devuelve o establece un [tipo de datos Single](#) que contiene la ubicación de la esquina izquierda de la ventana en la pantalla medida en [twips](#). Esta propiedad es de lectura y escritura.

Comentarios

El valor devuelto por la propiedad **Left** depende de si la ventana está [vinculada](#) o [acoplada](#).

Nota Cambiar el valor de la propiedad **Left** de una ventana vinculada o acoplada no tiene efecto mientras la ventana permanezca vinculada o acoplada.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

Lines (Propiedad)

Devuelve una cadena que contiene un número específico de líneas de código.

Sintaxis

object.**Lines**(*startline*, *count*) **As String**

La sintaxis de **Lines** tiene tres partes:

Parte	Descripción
-------	-------------

<i>object</i>	Requerido. Una expresión de objeto que evalúa un objeto de la lista Se aplica a.
<i>startline</i>	Requerido. Una longitud que especifica el número de línea en el que comenzar.
<i>count</i>	Requerido. Una longitud que especifica el número de líneas que quiere devolver.

Comentarios

El [número de líneas](#) en un [módulo de código](#) comienza en 1.

LinkedWindowFrame (Propiedad)

Devuelve un objeto **Window** que representa el marco que contiene la ventana. Esta propiedad es de sólo lectura.

Comentarios

La propiedad **LinkedWindowFrame** le permite tener acceso al objeto que representa el [marco de ventana vinculada](#), que tiene distintas propiedades que la ventana o ventanas que contiene. Si la ventana no está vinculada, la propiedad **LinkedWindowFrame** devuelve **Nothing**.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

MainWindow (Propiedad)

Devuelve un objeto **Window** que representa la ventana principal del [entorno de programación](#) de Visual Basic. Esta propiedad es de sólo lectura.

Comentarios

Puede utilizar el objeto **Window** devuelto por la propiedad **MainWindow** para agregar o quitar [ventanas acopladas](#). También puede utilizar el objeto **Window** devuelto por la propiedad **MainWindow** para maximizar, minimizar, ocultar o restaurar la ventana principal del entorno de desarrollo de Visual Basic.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

Major (Propiedad)

Devuelve un [tipo de datos Long](#) que contiene el número mayor de versión de la [biblioteca de tipos](#) referenciada. Esta propiedad es de sólo lectura.

Comentarios

El número devuelto por la propiedad **Major** corresponde al número mayor de versión almacenado en la biblioteca de tipos en la que ha establecido la referencia.

Minor (Propiedad)

Devuelve un [tipo de datos Long](#) que indica el número menor de versión de la [biblioteca de tipos](#) referenciada. Esta propiedad es de sólo lectura.

Comentarios

El número devuelto por la propiedad **Minor** corresponde con el número menor de versión almacenado en la biblioteca de tipos en la que ha establecido la referencia.

Mode (Propiedad)

Devuelve un valor que contiene el modo del [proyecto](#) especificado. Esta propiedad es de sólo lectura.

Valores devueltos

La propiedad **Mode** devuelve estos valores:

Constante	Descripción
vbext_vm_Run	El proyecto especificado está en modo de ejecución.
vbext_vm_Break	El proyecto especificado está en modo de interrupción.
vbext_vm_Design	El proyecto especificado está en modo de diseño.

Name (Propiedad, modelo de objeto de complementos de VBA)

Devuelve o establece un [tipo de datos String](#) que contiene el nombre usado en el código para identificar a un objeto. Es de lectura y escritura en los objetos

VBProject y **VBComponent** y de sólo lectura en los objetos **Property** y **Reference**.

Comentarios

La siguiente tabla describe cómo el valor de la propiedad **Name** se aplica a diferentes objetos.

Objeto	Resultado de utilizar el valor de la propiedad Name
VBProject	Devuelve o establece el nombre del proyecto activo.
VBComponent	Devuelve o establece el nombre del componente. Se produce un error si intenta establecer la propiedad Name a un nombre ya usado o a un nombre no válido.
Property	Devuelve el nombre de la propiedad como aparece en Property Browser . Es el valor usado como índice en la colección Properties . No se puede establecer el nombre.
Reference	Devuelve el nombre de la referencia en el código. No se puede establecer el nombre.

El nombre predeterminado para objetos nuevos es el tipo de objeto más un entero único. Por ejemplo, el primer nuevo objeto Form es Form1, otro nuevo objeto Form es Form2 y el tercer control TextBox que puede crear en un formulario es TextBox3.

La propiedad **Name** de un objeto debe comenzar con una letra y puede tener un máximo de 40 caracteres. Puede incluir números y caracteres subrayados () pero no puede tener signos de puntuación o espacios. Los [formularios](#) y [módulos](#) no pueden tener el mismo nombre como estos otros objetos públicos: **Clipboard**, **Screen** o **App**. Aunque el valor de la propiedad **Name** puede ser una [palabra clave](#), el nombre de la propiedad o el nombre de otro objeto puede crear conflictos en su código.

NumIndices (Propiedad, modelo de objeto de complementos de VBA)

Devuelve el número de índices de la [propiedad](#) devuelta por el objeto **Property**.

Comentarios

El valor de la propiedad **NumIndices** puede ser un entero de 0 a 4. En la mayoría de las propiedades, **NumIndices** devuelve 0. Las propiedades indexadas convencionales devuelven 1. Las [matrices](#) de la propiedad pueden devolver 2.

Object (Propiedad)

Devuelve o establece el valor de un objeto devuelto por una [propiedad](#). Esta propiedad es lectura y escritura.

Comentarios

Si la propiedad devuelve un objeto, debe utilizar la propiedad **Object** para devolver o establecer el valor de ese objeto.

Parent (Propiedad)

Devuelve el objeto o la [colección](#) que contiene otro objeto o colección. Esta propiedad es de sólo lectura.

Comentarios

La mayoría de los objetos tienen una propiedad **Parent** o **Collection** que apunta al objeto padre del objeto en ese modelo de objeto. La propiedad **Collection** se utiliza si el objeto primario es una colección.

Utilice la propiedad **Parent** para tener acceso a las [propiedades](#), [métodos](#) y [controles](#) de un objeto primario de un objeto.

Protection (Propiedad)

Devuelve un valor que indica el estado de protección de un [proyecto](#). Esta propiedad es de sólo lectura.

Valores devueltos

La propiedad **Protection** devuelve estos valores:

Constante	Descripción
vbext_pp_locked	El proyecto especificado está bloqueado.
vbext_pp_none	El proyecto especificado no está protegido.

ReferencesEvents (Propiedad)

Devuelve el objeto **ReferencesEvents**. Esta propiedad es de sólo lectura.

Valores

Los valores para el [argumento](#) que pasan a la propiedad **ReferencesEvents** son:

Argumento	Descripción
<i>vbproject</i>	Si <i>vbproject</i> señala a Nothing , el objeto devuelto suministrará eventos a la colección References de todos los objetos

VBProject en la colección **VBProjects**.

Si *vbproject* apunta a un objeto **VBProject** válido, el objeto devuelto suministrará eventos sólo en la colección **References** de ese [proyecto](#).

Comentarios

La propiedad **ReferencesEvents** toma un argumento y devuelve un [objeto origen de evento](#). El objeto **ReferencesEvents** es el origen de los eventos que se activan cuando se agregan o quitan referencias.

Saved (Propiedad)

Devuelve un valor [Booleano](#) que indica si el objeto ha sido modificado o no desde la última vez que se guardó. Lectura/escritura.

Valores devueltos

La propiedad **Saved** devuelve los siguientes valores:

Valor	Descripción
True	El objeto no ha sido modificado desde la última vez que se guardó.
False	El objeto ha sido modificado desde la última vez que se guardó.

Comentarios

El método **SaveAs** establece la propiedad **Saved** a **True**.

Nota Si establece la propiedad **Saved** a **False** a través de código, devuelve **False** y el objeto es marcado como si se hubiera modificado desde la última vez que se guardó.

SelectedVBComponent (Propiedad)

Devuelve el componente seleccionado. Sólo lectura.

Comentarios

La propiedad **SelectedVBComponent** devuelve el componente seleccionado en la [ventana Proyecto](#). Si el elemento seleccionado en la ventana **Proyecto** no es un componente, **SelectedVBComponent** devuelve **Nada**.

Top (Propiedad)

Devuelve o establece un [tipo de datos Single](#) que especifica la ubicación de la parte superior de la ventana en la pantalla medido en [twips](#). Esta propiedad es de lectura y escritura.

Comentarios

El valor devuelto por la propiedad **Top** depende de si la ventana está o no [acoplada](#), [vinculada](#) o en modo de ver anclado.

Nota Cambiar el valor de la propiedad **Top** de una ventana vinculada o acoplada no tiene efecto mientras la ventana permanezca vinculada o acoplada.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

TopLine (Propiedad)

Devuelve un [tipo de datos Long](#) que especifica el número de la línea de la parte superior del [panel de código](#) o establece la línea mostrada en la parte superior del panel de código. Esta propiedad es de lectura y escritura.

Comentarios

Utilice la propiedad **TopLine** para devolver o establecer la línea mostrada en la parte superior del panel de código. Por ejemplo, si quiere que la línea 25 sea la primera línea que se muestre en el panel de código, establezca la propiedad **TopLine** a 25.

El valor de la propiedad **TopLine** debe ser un número positivo. Si el valor de la propiedad **TopLine** es mayor que el número de líneas actual del panel de código, el valor será la última línea del panel de código.

Type (Propiedad, modelo de objeto de complemento de VBA)

Devuelve un valor numérico o de cadena que contiene el tipo de objeto. Esta propiedad es de sólo lectura.

Valores devueltos

Los valores de la propiedad **Type** para el objeto **Window** se muestran en la siguiente tabla:

Constante	Valor	Descripción
vbext_wt_CodeWindow	0	Ventana Código
vbext_wt_Designer	1	Diseñador
vbext_wt_Browser	2	Explorador de objetos
vbext_wt_Immediate	5	Ventana Inmediato
vbext_wt_ProjectWindow	6	Ventana Proyecto
vbext_wt_PropertyWindow	7	Ventana Propiedades
vbext_wt_Find	8	Cuadro de diálogo Buscar
vbext_wt_FindReplace	9	Cuadro de diálogo Buscar y reemplazar
vbext_wt_LinkedWindowFrame	11	Marco de ventana vinculada
vbext_wt_MainWindow	12	Ventana principal
vbext_wt_Watch	3	Ventana Inspección
vbext_wt_Locals	4	Ventana Locales
vbext_wt_Toolbox	10	Cuadro de herramientas
vbext_wt_ToolWindow	15	Ventana Herramienta

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin

embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

Los valores de la propiedad **Type** para el objeto **VBComponent** se muestran en la siguiente tabla:

Constante	Valor	Descripción
vbext_ct_StdModule	1	Standard module
vbext_ct_ClassModule	2	Class module
vbext_ct_MSForm	3	Microsoft Form
vbext_ct_ActiveXDesigner	11	ActiveX Designer
vbext_ct_Document	100	Document Module

Los valores de la propiedad **Type** para el objeto **Reference** se muestran en la siguiente tabla:

Constante	Valor	Descripción
vbext_rt_TypeLib	0	Biblioteca de tipos
vbext_rt_Project	1	Proyecto

Los valores de la propiedad **Type** para el objeto **VBProject** se describen en la tabla siguiente:

Constante	Valor	Descripción
vbext_pt_HostProject	100	Proyecto host
vbext_pt_StandAlone	101	Proyecto individual

Value (Propiedad)

Devuelve o establece un [tipo de datos Variant](#) que especifica el valor de la [propiedad](#). Esta propiedad es de lectura y escritura.

Comentarios

Puesto que la propiedad **Value** devuelve un tipo de datos **Variant**, puede tener acceso a cualquier propiedad. Para tener acceso a una lista, utilice la propiedad **IndexedValue**.

Si la propiedad que representa el objeto **Property** es de lectura y escritura, la propiedad **Value** es de lectura y escritura. Si la propiedad es de sólo lectura, se produce un error al intentar establecer la propiedad **Value**. Si la propiedad es de sólo escritura, se produce un error al intentar devolver la propiedad **Value**.

La propiedad **Value** es la propiedad predeterminada para el objeto **Property**.

VBE (Propiedad)

Devuelve la raíz del objeto **VBE**. Esta propiedad es de sólo lectura.

Comentarios

Todos los objetos tienen una propiedad **VBE** que apuntan a la raíz del objeto **VBE**.

Version (Propiedad)

Devuelve un [tipo de datos String](#) que contiene la versión de Visual Basic para Aplicaciones que la aplicación esta usando. Esta propiedad es de sólo lectura.

Comentarios

El valor de la propiedad **Version** es una cadena que comienza con uno o dos dígitos, un periodo y dos dígitos; el resto de la cadena no está definida y puede contener texto o números.

Visible

(Propiedad)

En el objeto **Window**, devuelve o establece un valor de [tipo de datos Boolean](#) que especifica la visibilidad de una ventana. Es de lectura y escritura. Para el objeto **CodePane** devuelve un valor **Boolean** que indica si el [panel de código](#) es o no visible en la ventana. Es de sólo lectura.

Valores devueltos

La propiedad **Visible** devuelve los siguientes valores:

Valor	Descripción
True	(Predeterminado) El objeto es visible.
False	El objeto está oculto.

Width (Propiedad)

Devuelve o establece un [tipo de datos Single](#) que contiene el ancho de la ventana medido en [twips](#). Esta propiedad es de lectura y escritura.

Comentarios

Cambiar el valor de la propiedad **Width** de una [ventana vinculada](#) o de una [ventana acoplada](#) no tiene efecto mientras la ventana permanezca vinculada o acoplada.

Importante Los objetos, propiedades y métodos para controlar ventanas vinculadas, marcos de ventanas vinculadas y ventanas acopladas se incluyen en Macintosh por motivos de compatibilidad con el código escrito en Windows. Sin embargo, estos elementos de lenguaje generarán errores en tiempo de ejecución cuando se ejecuten en Macintosh.

Window (Propiedad)

Devuelve la ventana en la que se muestra el [panel de código](#). Esta propiedad es de sólo lectura.

WindowState (Propiedad)

Devuelve o establece un valor numérico que especifica el estado visual de la ventana. Esta propiedad es de lectura y escritura.

Valores

La propiedad **WindowState** devuelve o establece los siguientes valores:

Constante	Valor	Descripción
vbext_ws_Normal	0	(Predeterminado) Normal
vbext_ws_Minimize	1	Minimizado (minimizado a un icono)
vbext_ws_Maximize	2	Maximizado (ampliada hasta el tamaño máximo)

AddIns (colección)



Devuelve una colección de complementos registrados por VBA.

Sintaxis

object.**AddIns**

Comentarios

La colección **AddIns** es accesible a través del objeto **VBE**. Cada complemento enumerado en el Administrados de complementos en VBE.tiene un objeto en la colección **AddIns**.