



aprenderaprogramar.com

# Tipos de errores en Visual Basic: de compilación, de ejecución, de sintaxis. Depuración. (CU00352A)

Sección: Cursos

Categoría: Curso Visual Basic Nivel I

Fecha revisión: 2029

Autor: Mario R. Rancel

Resumen: Entrega nº51 del Curso Visual Basic Nivel I

29

## ERRORES

Si has venido trabajando los diferentes programas y ejercicios que hemos planteado a lo largo del curso se te habrán presentado numerosos mensajes de error. Los mensajes de errores interceptados por Visual Basic son de dos tipos: "Errores de compilación" y "Errores de ejecución", coincidentes con la clasificación de errores que se hace en el curso "Bases de la programación nivel II" de aprenderaprogramar.com.

Ya hemos hablado algo de errores de sintaxis, y de la ayuda que nos proporciona Visual Basic a través de la corrección automática de errores. Los errores de sintaxis serán muy frecuentes y quizás incómodos, pues casi siempre nos encontraremos con uno o varios al tratar de ejecutar un programa. Sin embargo, son los menos "peligrosos", ya que de su fácil detección se deriva su corrección segura. Prueba a ejecutar este código donde existe un error de sintaxis (Lbel en vez de Label):

### Código versiones menos recientes de VB:

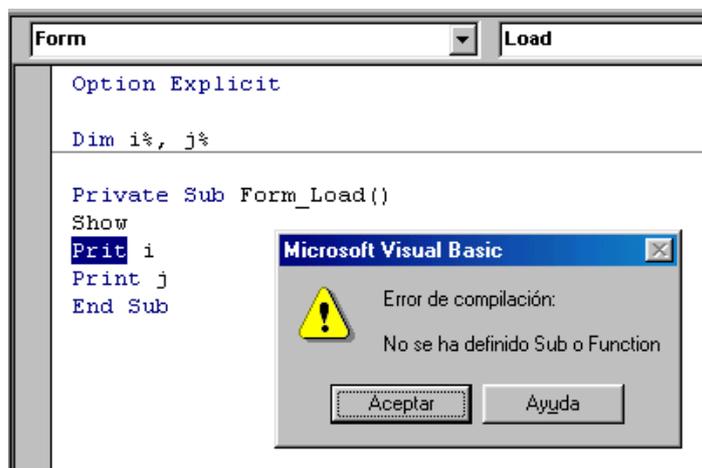
```
'Curso VB aprenderaprogramar.com
Option Explicit
Dim i%, j%

Private Sub Form_Load()
Lbel1 = i
Label1 = Label1 & j
End Sub
```

### Código versiones más recientes de VB:

```
REM Curso Visual Basic aprenderaprogramar.com
Option Explicit On
Public Class Form1
Private Sub Form1_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    Dim i%, j%
    Lbel1.text = i
    Label1.Text = Label1.Text & j
End Sub
End Class
```

Al tratar de ejecutar el código te aparecerá un mensaje como "Error de compilación: variable no definida" ó "No se ha definido Sub o Function" ó "Lbl1 no está declarado" ó similar.



No se indica directamente que se trata de un error de sintaxis, ya que en algunas ocasiones a Visual Basic no le resulta fácil determinar si el error es de sintaxis o de otro tipo. En este caso por ejemplo, se interpreta que se está invocando una variable o procedimiento de nombre Lbl y te indica que dicho procedimiento o variable no está definida y por tanto no se puede invocar. Te aparece subrayado el término que está dando lugar al error, con lo cual rápidamente podrás comprobar que se trata de una escritura indebida y proceder a su corrección.

Los errores por procesos no válidos son en general detectados al tratar de ejecutar los programas, indicándonos Visual Basic qué tipo de error se ha producido, y en qué línea ha tenido lugar, aunque a veces no se detectan y dan lugar a “resultados extraños”. Prueba a ejecutar el siguiente código:

**Código versiones menos recientes de VB:**

```
'Curso VB aprenderaprogramar.com
Option Explicit
Dim i%, j%

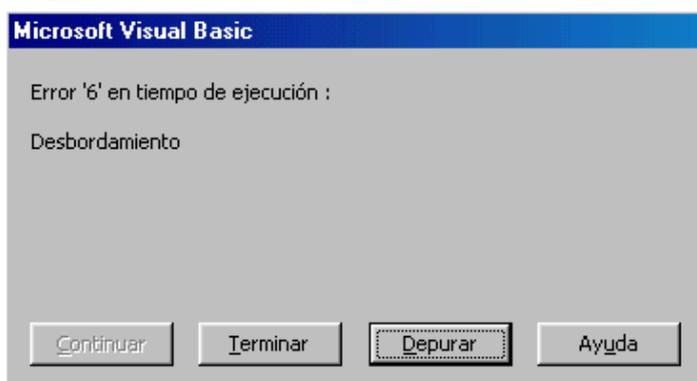
Private Sub Form_Load()
Show
Label1 = i
Label1 = j
Label1 = i / j
End Sub
```

**Código versiones más recientes de VB:**

```
REM Curso Visual Basic aprenderaprogramar.com
Option Explicit On
Public Class Form1
Private Sub Form1_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    Dim i%, j%
    Lbell.text = i
    Label1.Text = Label1.Text & j
End Sub
End Class
```

Al llegar a la línea donde se trata de mostrar el valor de  $i / j$  el ordenador se encuentra con una indeterminación matemática ya que cero dividido entre cero no es un valor calculable. Esto puede dar lugar a dos tipos de situaciones:

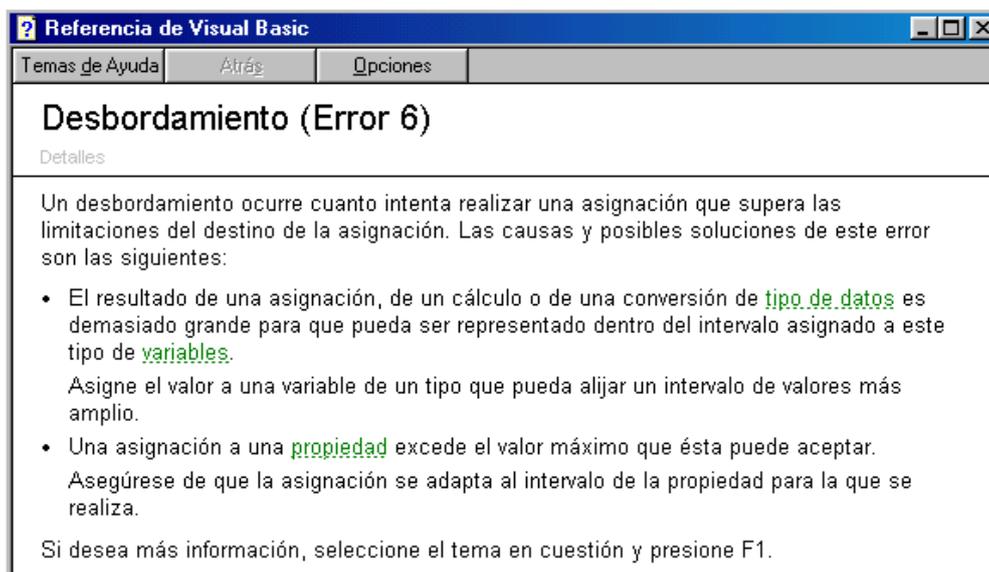
- La detención del programa y aparición de un mensaje de error.



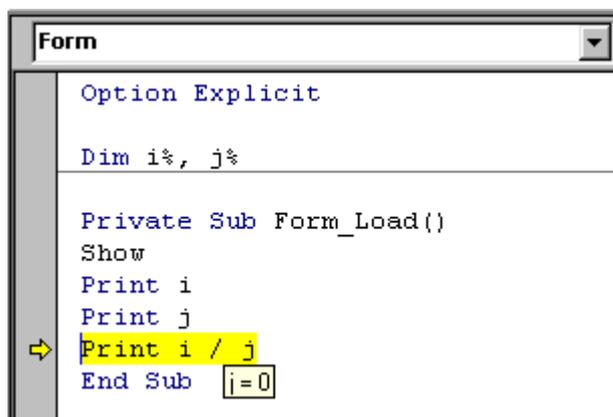
- La obtención de un resultado extraño, por ejemplo que por pantalla aparezca NeuN (este valor extraño indica “resultado matemáticamente no determinable” y posiblemente termine generando un error y detención de un programa cuando surge en programas largos que utilizan este resultado para realizar más operaciones adicionales).

Cuando la ejecución de un programa se detiene debido a la aparición de un error Visual Basic nos suele ofrecer ayuda o información sobre el error que nos sirva para realizar las correcciones oportunas. Una vez detenido el programa podemos pulsar el botón “Terminar” para concluir la ejecución o también podemos pulsar en ayuda o sobre el error en una lista de errores para obtener ayuda e información adicional. La opción de ayuda nos conducirá a una descripción del tipo de error, recomendaciones para evitarlo, etc.

Podemos intervenir sobre el código para tratar de subsanar el error, con posibilidad de proseguir posteriormente la ejecución (pulsando F5). Visual Basic nos resalta y señala con una flecha la línea en que se ha producido la detención del programa, lo cual nos sirve de ayuda para corregir el error.



Cuando el programa está detenido podemos obtener información posicionando el cursor sobre un elemento del código (por ejemplo para conocer el valor que tiene una variable o el tipo de una variable).



No vamos a extendernos sobre las posibilidades que sobre errores ofrece Visual Basic, como programadores deberemos ir conociéndolas poco a poco a medida que tengamos más experiencia.

En relación a errores lógicos, las posibilidades son muy variadas. Algunos errores lógicos pueden ser detectados por Visual Basic, pero no conviene tener demasiadas expectativas al respecto. No te vamos a pedir que pruebes el siguiente código ya que puede bloquear el computador:

**Código versiones menos recientes de VB:**

```
'Curso VB aprenderaprogramar.com
Option Explicit

Private Sub Form_Load()
Do

Loop

End Sub
```

**Código versiones más recientes de VB**

```
REM Curso Visual Basic aprenderaprogramar.com
Option Explicit On
Public Class Form1
    Private Sub Form1_Load(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        Do

            Loop
        End Sub
    End Class
```

Pero si por casualidad estás construyendo un programa y te has dejado olvidado o "sin solucionar" un bucle de este tipo, lo que puedes esperar es que el ordenador se quede "colgado" (bloqueado repleto de un bucle continuamente). Con un poco de suerte podrás finalizar el programa sin necesidad de reiniciar, perdiendo la información que no hubieras guardado.

Por ello te recomendamos como medida de prudencia que antes de ejecutar cualquier programa, y en especial desarrollos extensos, procedas a guardar el proyecto de programa a través de la opción Guardar de Visual Basic, o bien que transfieras el código a un procesador de textos y procedas a guardarlo como texto. De esta manera, ante bloqueos inesperados, podrás recuperar el trabajo realizado y proceder a corregirlo.



Existen diversos tipos de errores interceptables por Visual Basic.

Puedes consultar una lista de ellos en la ayuda del programa buscando "Errores interceptables". De esta lista, compuesta por más de 60 tipos de error, podemos extraer información relativa a los tipos de errores detectables, es decir, aquellos cuya ocurrencia de alguna manera podemos tratar a través de instrumentos de gestión de errores (On Error, objeto Err...).

Tratar de estudiar todos los tipos de error no tiene demasiado sentido. Más bien se tratará de ir familiarizándonos con ellos a través de nuestra interacción con el Visual Basic. Así pues, cuando se nos presente un error del tipo: "No hay suficiente espacio de pila", podemos ver a través de la ayuda cuáles son las causas de este error que afecta a la pila (espacio de memoria temporal). Por ejemplo que nuestro programa cuente con procedimientos anidados muy profundamente, cuestión frecuente con los procedimientos recursivos (es decir, los procedimientos que se llaman a sí mismos).

En cuestión de errores se trata de prevenir problemas pero también de buscar soluciones. Ya lo dice el refrán: "Burro cargado busca camino", o aplicado al ámbito de la programación "Programador con problemas busca soluciones".

**Próxima entrega: CU00353A**

**Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:**  
[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=37&Itemid=61](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=37&Itemid=61)