



aprenderaprogramar.com

# If Then Else con Visual Basic (condicionales). Sintaxis de bloque o línea. Ejemplos resueltos (CU00323A)

Sección: Cursos

Categoría: Curso Visual Basic Nivel I

Fecha revisión: 2029

Autor: Mario R. Rancel

Resumen: Entrega nº22 del Curso Visual Basic Nivel I

29

## IF ... THEN (SI ... ENTONCES) E IF ... THEN ... ELSE (SI ... ENTONCES ... SINO)

La sintaxis a emplear con Visual Basic es muy similar a la que se estudia en los cursos de algoritmia y pseudocódigo, con la sustitución del Si por If, del Entonces por Then, del SiNo por Else y del Finsi por End If. La única salvedad a tener en cuenta es que Visual Basic permite además de la escritura en bloques la escritura en una sola línea.

Sintaxis en bloque:

```
If [condición] Then
    Instrucción 1
    Instrucción 2
    .
    .
    .
    Instrucción n
Else
    Instrucción A
    Instrucción B
    .
    .
    .
    Instrucción Z
End If
```

Sintaxis en una sola línea:

```
If [condición] Then Instrucción 1 Else Instrucción A
```

También es posible encadenar varias acciones derivadas de un Then o de un Else en una sola línea. Sin embargo, recomendamos usar la opción de sintaxis en una sola línea exclusivamente para desarrollos sencillos como puede ser una instrucción en caso de que se cumpla la condición y otra en caso de que no se cumpla. Si intervienen varias instrucciones, siempre resultará más legible usar el formato de bloque.

La instrucción If ... Then es anidable dentro de sí misma, siendo siempre preferible el formato de bloque cuando se vayan a generar anidamientos.

**Ejemplo:**

```
If A > 7 And B < 5 Then
    MsgBox("A es mayor que 7 y B menor que 5")
End If
```

También válido es:

```
If A > 7 And B < 5 Then MsgBox ("A es mayor que 7 y B menor que 5")
```

Si introducimos una cláusula Else el modelo es:

```
If A > 7 And B < 5 Then
    MsgBox ("A es mayor que 7 y B menor que 5")
Else
    MsgBox ("A no es mayor que 7 ó B no es menor que 5")
End If
```

También válido usar el símbolo \_ (guión bajo) para indicar continuación de línea:

```
If A > 7 And B < 5 Then MsgBox ("A es mayor que 7 y B menor que 5") _
Else MsgBox "A no es mayor que 7 ó B no es menor que 5"
```

Una cuestión a tener en cuenta es que cuando la condición indicada por un If en bloque no se cumple, se produce el salto a la siguiente instrucción inmediatamente posterior a la palabra End If. En cambio, cuando la condición indicada por un If en una línea no se cumple, se produce el salto a la siguiente línea ignorando todas las instrucciones que pudieran encontrarse en la misma línea del If. Veámoslo con un ejemplo. Ejecuta estos dos pequeños programas:

#### Programa 1 (versiones menos recientes VB):

```
Rem Curso Visual Basic aprenderaprogramar.com

Option Explicit
Dim a%

Private Sub Form_Load()
a = 3
If a = 5 Then
    MsgBox("Se cumple que a vale 5")
End If
a = 25
MsgBox("El valor actual de a es " & a)
End Sub
```

#### Programa 1 (versiones más recientes VB):

```
REM Curso Visual Basic aprenderaprogramar.com
Option Explicit On
Public Class Form1
    Dim a%

    Private Sub Form1_Load(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        a = 3
        If a = 5 Then
            MsgBox("Se cumple que a vale 5")
        End If
        a = 25
        MsgBox("El valor actual de a es " & a)
    End Sub
End Class
```

**Programa 2 (versiones menos recientes VB):**

```
Rem Curso Visual Basic aprenderaprogramar.com
Option Explicit
Dim a%

Private Sub Form_Load()
a = 3
If a = 5 Then MsgBox ("Se cumple que a vale 5"): a = 25
MsgBox ("El valor actual de a es " & a)
End Sub
```

**Programa 2 (versiones más recientes VB):**

```
REM Curso Visual Basic aprenderaprogramar.com
Option Explicit On
Public Class Form1
    Dim a%

    Private Sub Form1_Load(ByVal sender As
System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        a = 3
        If a = 5 Then MsgBox("Se cumple que a
vale 5") : a = 25
        MsgBox("El valor actual de a es " & a)
    End Sub
End Class
```

En el programa 1 el resultado es que se muestra en pantalla "El valor actual de a es 25", mientras que en el programa 2 se muestra "El valor actual de a es 3". La razón para que en la versión 2 no se ejecute la orden a = 25 es que al no cumplirse la condición del If se produce un salto a la siguiente línea, ignorándose todas las instrucciones que se encuentren en la misma línea del If. En general consideramos recomendable habituarse a usar la sintaxis de bloque siempre pues resulta más fácil de interpretar y evita posibles confusiones como ésta, donde muchas personas podrían pensar que si no se cumple el if se ejecutará la instrucción que aparece en la misma línea a continuación.

**Próxima entrega: CU00324A**

**Acceso al curso completo en 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)