

Constantes y variables (Visual C# Express)

Visual Studio 2008 Personas que lo han encontrado útil: 11 de 24

Actualización: noviembre 2007

Una variable representa un valor numérico o de cadena o un objeto de una clase. El valor que la variable almacena puede cambiar, pero el nombre sigue siendo el mismo. Una variable es un tipo de *campo*. El código siguiente es un ejemplo sencillo de cómo declarar una variable de entero, asignarle un valor y, a continuación, asignarle un nuevo valor.

C#

```
int x = 1; // x holds the value 1
x = 2;    // now x holds the value 2
```

En C#, las variables se declaran con un tipo de datos y una etiqueta concretos. Si hasta ahora sólo ha utilizado lenguajes con tipos definidos de forma imprecisa como JScript, estará acostumbrado a emplear el mismo tipo "var" para todas las variables, pero en C# tiene que especificar si la variable es de tipo **int**, **float**, **byte**, **short** u otro cualquiera entre más de 20 tipos de datos diferentes. El tipo especifica, entre otras cosas, la cantidad de memoria exacta que se debe asignar para almacenar el valor cuando la aplicación se ejecuta. El lenguaje C# fuerza ciertas reglas al convertir una variable de un tipo en otro. Para obtener más información, vea [Tipos de datos integrados \(Visual C# Express\)](#).

C#

```
int answer = 42;
string greeting = "Hello, World!";
double bigNumber = 1e100;

System.Console.WriteLine("{0} {1} {2}", answer, greeting, bigNumber);
```

Constantes

Una constante es otro tipo de campo. Contiene un valor que se asigna cuando se compila el programa y nunca cambia después. Las constantes se declaran con la palabra clave `const`; son útiles para que el código sea más legible.

C#

```
const int speedLimit = 55;
const double pi = 3.14159265358979323846264338327950;
```

Una variable `readonly` es como una constante pero su valor se asigna al iniciar el programa. Esto le permite establecer el valor basándose en alguna otra condición que no conoce hasta que se ejecuta el programa. Pero después de esa primera asignación, el valor no puede cambiar de nuevo mientras el programa se está ejecutando.

Vea también

Conceptos

[Manual del lenguaje C#](#)

[Tipos de datos integrados \(Visual C# Express\)](#)

[Enumeraciones \(Visual C# Express\)](#)

[Tipos de datos integrados \(Visual C# Express\)](#)

Adiciones de comunidad

© 2014 Microsoft. Reservados todos los derechos.