FUNDAMENTOS DE VISUAL BASIC.NET

VB tiene una serie de variables, que permiten el manejo de datos. Podemos destacar los siguientes tipos de variables:

|  |  |  |  |
| --- | --- | --- | --- |
| TIPO | CLS | RANGO | TAMAÑO |
| Boolean | System.Boolean | True / False | 2 Bytes |
| Byte | System.Byte | 8 bits 0…255 | 1 Byte |
| (\*) Char | System.Char | 0…65535 (Sin signo) | 2 Bytes |
| Date | System.DateTime | 1/1/0001-31/12/9999 | 8 Bytes |
| Decimal (Nº entero extra largo) | System.Decimal | Nº 28 dígitos | 16 Bytes |
| Double | System.Double | Valores en coma flotante de 64 bits | 8 Bytes |
| Integer | System.Int32 | Números enteros | 4 Bytes |
| Long | System.Int64 | Nº enteros largos no tanto como decimal. | 8 Bytes |
| Object (Permite guardar un valor como si fuese objeto) | System.Object |  | 4 Bytes |
| Short (Enteros cortos) | System.Int16 | -32768 A 32767 | 2 Bytes |
| Single (Para representar núm. decimales) | System.Single | Coma flotante de 32 bit | 4 Bytes |
| String | System.String | 0 a 2000 Caracteres | Según Cantidad |
| Definidas por el usuario (Estructuras) | System.ValueTime | Según definición | Según definición |
|  |

Se destacan:

1. Núm. Enteros:
   1. Integer (Normales)
   2. Long (Largos)
2. Tipo cadena:
   1. Char (1 carácter)
   2. String (cadenas)
3. Decimales
   1. Single (Decimal normal)
   2. Double (Decimales muy grandes)

Definición de Variables y constantes con Visual Basic:

Variables: Posición de memoria que se le da un nombre y asigna un valor. VB tiene dos tipos de variables:

* Explicitas: Son aquellas variables que de forma deliberada es definida por el usuario para ser utilizada. Sintaxis:

Dim Static

Private Protected nomVariable as tipoVariable [= valor inicial]

Public Friend

ReadOnly

nomVariable: Es el nombre que va a identificar la variable, no puede tener . $%&@#… ni puede ser una palabra reservada.

tipoVariable: Que tipo de variable voy a utilizar.

Se puede poner también un valor inicial.

* + Dim: Se utiliza para definir variables de ámbito local referente a un procedimiento o función. Es el más utilizado.

Dim hola As String = “Hola Mundo!”

Dim fecha As Date = #10/03/2012#

Dim con1,con2 As Integer (no se pueden inicializar)

* + Private: Se utiliza para definir variables de ámbito local a nivel bloque, es decir a nivel estructuras repetitivas o condicionales.
  + Public: Se utiliza para definir variables de tipo global, pero su ámbito global estará determinado del lugar donde se defina.
  + ReadOnly: Variables que no pueden ser modificadas, solo leídas, no se puede escribir en ellas.
  + Static: Permite definir una variable cuyo valor conserva valores anteriores.
  + Protected: Se trata de una variable protegida para ser utilizada en una clase concreta, y su ámbito es la clase donde se define.
  + Friend: Se trata de una definición de variables de ámbito global pero que solo pueden ser utilizadas en el ensamblado propio del programa, evitando que las utilice otro programa.

* Implícitas: VB no obliga a declarar explícitamente las variables. Cuando VB define variables automáticamente del tipo valor que se le asigne a esta es una variable implícita. Para obligar a realizar definición de las variables de forma explícita, será necesario acceder a las propiedades del proyecto y jugar con on/off la propiedad option explicit, que está en propiedades/proyecto/solapa/compilar.

Constantes: Valor asignado a una variables que no cambia durante un proceso. Hay dos tipos de constantes:

* ‘Simbólicas’: Son aquellas constantes que están definidas por el usuario, y que lleva la palabra reservada const.

Dim const a As Integer = 120

(Siempre tiene que estar definida)

* ‘Intrínsecas’: Son una serie de constantes internas de VB que Microsoft nos deja manipular.

AMBITO DE LAS VARIABLES:

Se pueden destacar cuatro ámbitos:

1. Ámbito de bloque: Hace referencia a nivel de estructura repetitiva (for, while…), que pueden ser Dim, Private, Static.
2. Ámbito de procedimiento o función: Consiste en que su ámbito es el procedimiento donde se ha definido. Todas las estructuras repetitivas tienen ese ámbito. Se pueden declarar Dim y Static.
3. Ámbito de namespace: Es una forma global al formulario (clase), por lo que la única forma de hacerlo es Public.
4. Ámbito de módulo: Consiste en definir una variable global solo con Public y cuyo ámbito es toda la aplicación, es decir, todos los procedimientos de todos los formularios de la aplicación. Se crea un modulo, que se inserta en nuestro proyecto, definiendo todas las variables que se necesiten y se pueden utilizar en todos los formularios.

EXPRESIONES Y OPERADORES:

Una expresión es una secuencia de operandos y operadores. .Net tiene los siguientes operadores:

* Aritméticos (Según importancia):
  + ^ (potencia)
  + \* /
  + Mod (Para saber el resto de una división).
  + + -
* Relacionales:
  + =
  + < o >
  + <> (distinto)
  + <=
  + >=
  + LIKE (para comparar y buscar cadenas).
* Logicos:
  + NOT (no)
  + AND (y)
  + OR (o)
* De cadena:
  + + (concatenación)
  + &: Se utiliza como el + en Java. Ejemplo: (“El nombre es “&name).