



PROGRAMACIÓN EN C# .NET

Introducción a la Programación en C# .NET

Ing. Bruno López Takeyas
Instituto Tecnológico de Nuevo Laredo

0

1

CONTENIDO

1. Introducción a la Programación en C# .NET
2. Declaraciones
3. Instrucciones fundamentales
4. Arreglos
5. Métodos y parámetros
6. Tipos de datos definidos por el usuario
7. Archivos

0

2

MATERIAL DISPONIBLE

- <http://ramonroque.com/Materias/pooTec.htm>
- Libro: El lenguaje de Programación C#
- Autor: José Antonio González Seco
- <http://www.josanguapo.com/>

3

BIBLIOGRAFÍA

- Archer, Tom. "A fondo C#". Editorial McGraw Hill. 2001.
- Charte, Francisco. "Manual de Introducción a Microsoft Visual C# 2005 Express Edition".
- Deitel & Deitel. "Programming in C#". Editorial Prentice Hall.
- Ferguson, Jeff. "La Biblia de C#". Editorial Anaya. 2003.
- Miles, Rob. "C# Development". Department of Computer Science. University of Hull. 2008-2009.

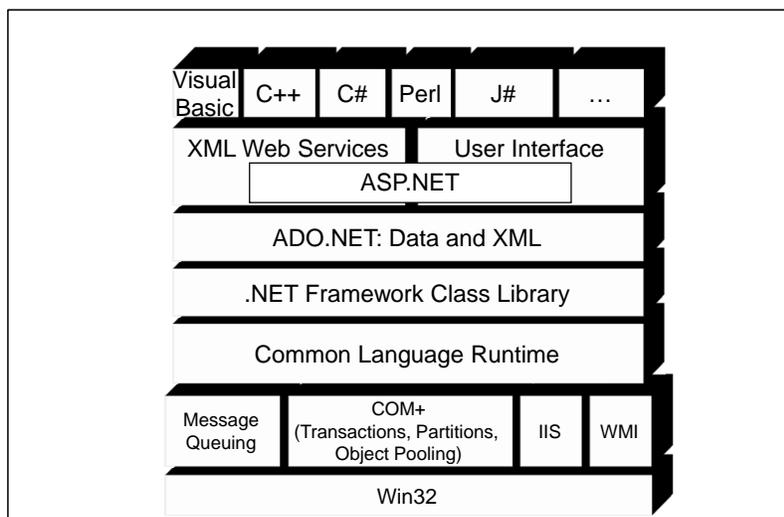
4

1.- INTRODUCCIÓN A LA PROGRAMACIÓN EN C# .NET

- Introducción a la plataforma Microsoft .NET
 - El framework .NET y sus componentes
 - Principales clases del framework .NET
- Microsoft Visual Studio .NET
 - Compilación y ejecución de programas
 - El debugger
 - Breakpoints
 - Watches
 - Ejecutar paso a paso
 - Examinar variables en tiempo de ejecución

5

INTRODUCCIÓN A LA PLATAFORMA MICROSOFT .NET



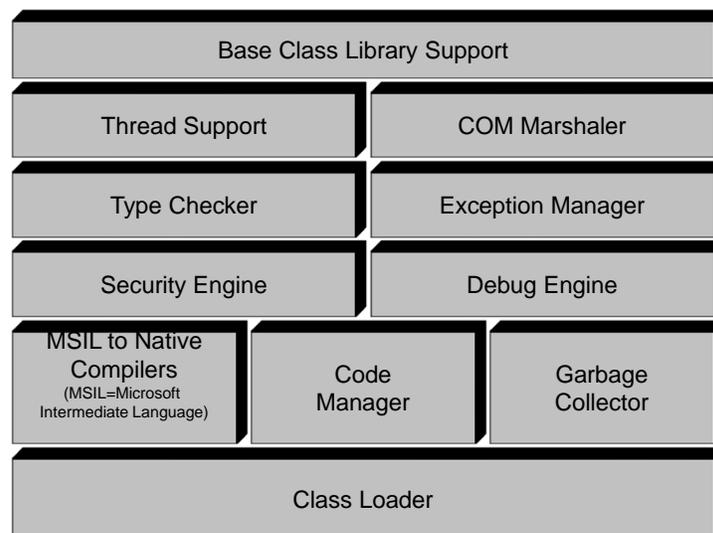
6

COMPONENTES DEL FRAMEWORK .NET

- Common Language Runtime
- Librerías de clases del Framework .NET
- ADO.NET: Datos and XML
- Formas Web y Servicios Web XML
- Interfaces para Windows

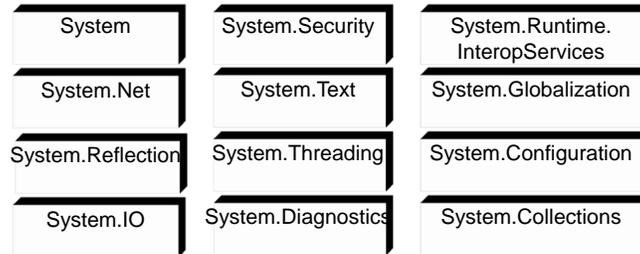
7

COMMON LANGUAGE RUNTIME



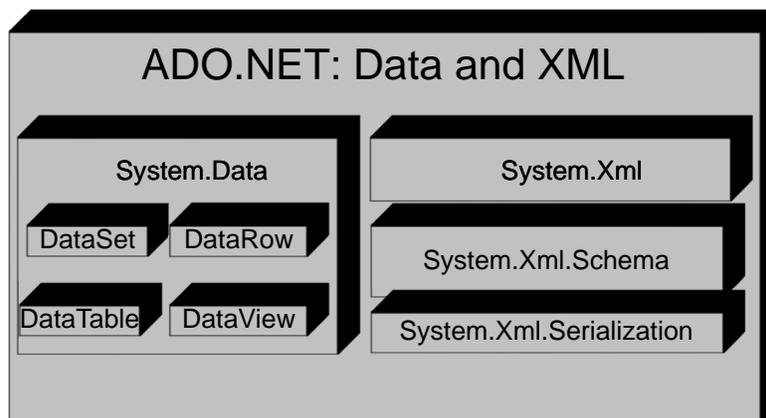
8

LIBRERÍAS DE CLASES DEL FRAMEWORK .NET



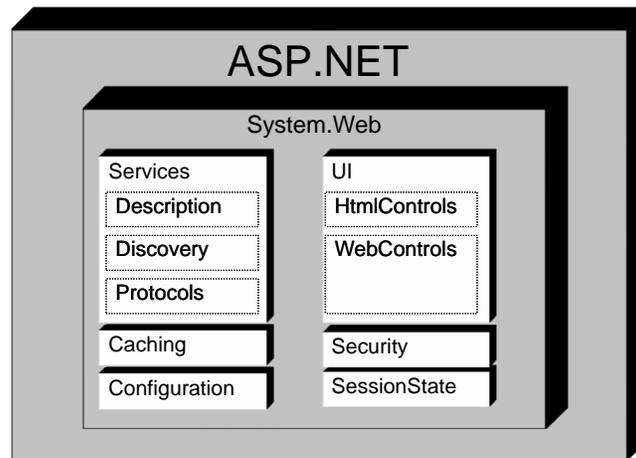
9

ADO .NET: DATOS Y XML



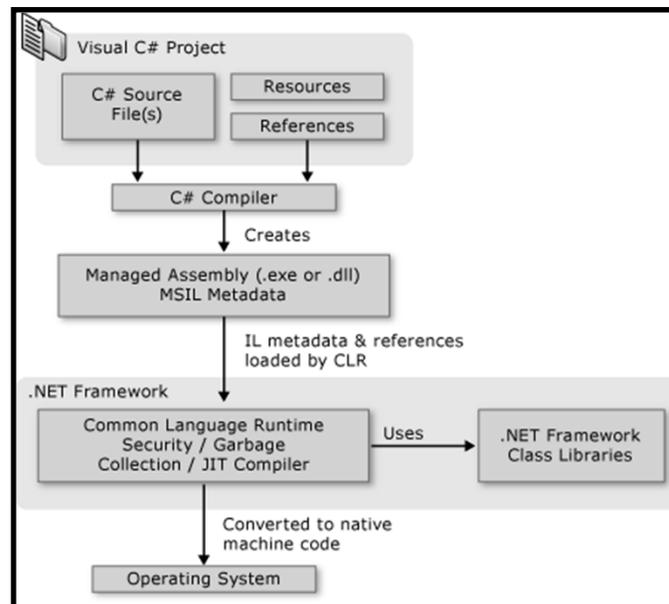
10

FORMAS WEB Y SERVICIOS WEB XML



11

COMPILACIÓN EN C#



12

LECTURAS ADICIONALES

Capítulo	Tema	Libro	Autor	Págs.
2	Introducción a Microsoft .NET	C# a fondo	Tom Archer	21 - 31
1	Introducción a Microsoft .NET	El lenguaje de Programación C#	José Antonio González Seco	10 -21
1	Introducción a C# .NET	Filminas	Ing. Ramón Roque Hernández	

Se recomiendan estas lecturas para dominar el ambiente del Framework .NET

13

HERRAMIENTAS DE DESARROLLO

- Descargar Microsoft Visual Studio Versión Express en:
 - <http://www.microsoft.com/express/Downloads/#2010-Visual-CS/>
- Microsoft .NET Framework SDK
 - msdn.microsoft.com/net/
- Mas información
 - <http://www.microsoft.com/net/>
- Sharp Develop
 - <http://www.icsharpcode.net>

14

Microsoft Visual Studio .NET



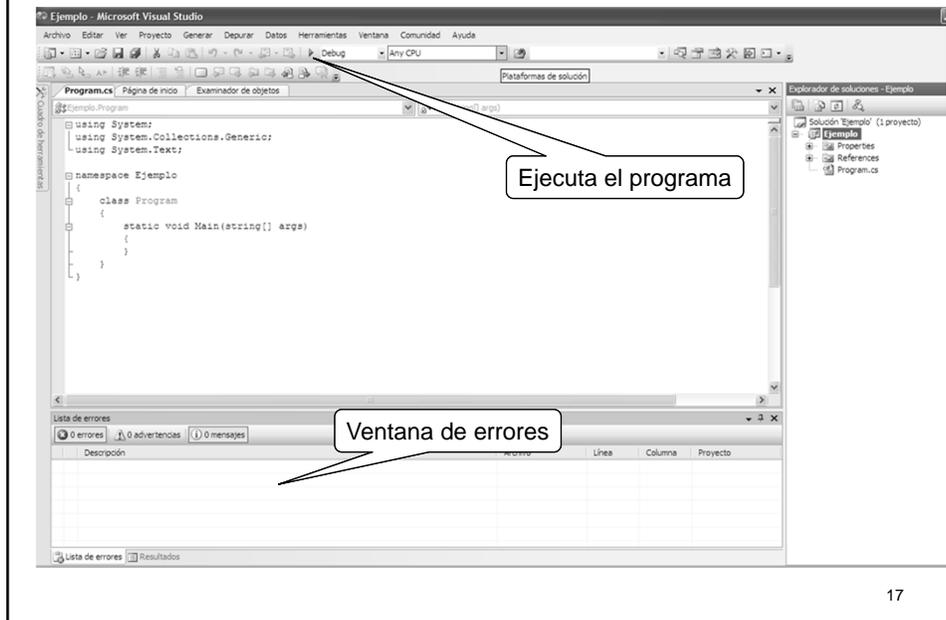
15

Crear un nuevo proyecto



16

EJECUTANDO UN PROGRAMA



LECTURAS ADICIONALES

Capítulo	Tema	Libro	Autor	Págs.
1	Primeros pasos	Manual de Microsoft Visual Studio 2005. Express Edition	Francisco Charte Ojeda	9 - 26

Se recomiendan estas lecturas para dominar el uso del Microsoft Visual Studio 2005

MÉTODOS DE LA CONSOLA

- `Console.Clear(); // Limpia la pantalla`

- Para desplegar mensajes en la pantalla:
 - `Console.Write();`
 - `Console.WriteLine();`

- Para capturar datos desde el teclado:
 - `Console.Read();`
 - `Console.ReadLine();`

19

MÉTODOS Write()

- Para desplegar mensajes en la pantalla:
 - `Console.Write("Hola mundo");`
 - `// No provoca Carry Return`

 - `Console.WriteLine("Hola mundo");`
 - `// Provoca Carry Return`

 - `Console.Write("La edad de {0} es {1}", nombre, edad);`

20

FORMATO NUMÉRICO

■ { N, M : Formato }

- N = Número de parámetro
- M = Ancho y justificación del dato
- Formato = Especifica cómo se mostrará el dato numérico

Formato	Descripción
C	Moneda (currency)
D	Decimal entero
E	Exponencial (notación científica)
F	Flotante
N	Despliega el número con comas separadoras
X	Hexadecimal
P	Porcentaje

21

FORMATO NUMÉRICO

Sentencia	Salida
<code>Console.WriteLine(" {0:C} ", 12345.6789);</code>	\$12,345.68
<code>Console.WriteLine(" {0:D5} ", 88);</code>	00088
<code>Console.WriteLine(" {0:E} ", 888.8);</code>	8.888000E+002
<code>Console.WriteLine(" {0:F3} ", 888.8888);</code>	888.889
<code>Console.WriteLine(" {0,8:F2} ", 145.3);</code>	145.30
<code>Console.WriteLine(" {0:N} ", 8888888.8);</code>	8,888,888.80
<code>Console.WriteLine(" {0:X4} ", 88);</code>	0058
<code>Console.WriteLine(" {0:P} ", 0.12345);</code>	12.35%

22

SECUENCIAS DE ESCAPE

Secuencia de escape	Descripción
\n	New line (cambio de línea)
\t	Tabulador horizontal
\r	Carriage return (Retorno de carro)
\\	Diagonal invertida
\"	Comillas
\'	Apóstrofe
\a	Alerta (produce un sonido)
\b	Backspace
\f	Form feed (cambio de página)
\0	Nulo

23

CARACTERES UNICODE

- El Unicode proporciona un estándar de codificación de caracteres y símbolos a nivel mundial
- Incluye el conjunto de caracteres ASCII
- En ASCII, cada carácter requiere 1 byte (que contiene 0's y 1's) y se representa con un número entre 0 y 255; por lo tanto, sólo tiene 256 caracteres (representa una parte del conjunto de símbolos a nivel mundial)

24

CARACTERES UNICODE (cont)

- El Estándar Unicode codifica todos los caracteres y símbolos en un rango de 0 a 10FFFF hexadecimal
- Consultar www.unicode.org
- Existen 3 formatos:
 - UTF-8 (8 bits)
 - UTF-16 (16 bits)
 - UTF-32 (32 bits)
- Visual Studio .NET usa UTF-16 para representar todos los caracteres

25

USO DE CARACTERES UNICODE

- Usar `\u` y el código hexadecimal del símbolo
 - `Console.WriteLine("\u00D1");` // Ñ
 - `Console.WriteLine("\u00E1");` // á
 - `Console.WriteLine("\u00C1");` // Á
 - `Console.WriteLine("\u00DF");` // ß
 - `Console.WriteLine("\u00BE");` // ¾
 - `Console.WriteLine("\u2557");` // ┐
 - `Console.WriteLine("\u263A");` // ☺
 - `Console.WriteLine("\u2660");` // ♠

26

LECTURAS ADICIONALES

Capítulo	Tema	Libro	Autor	Págs.
Apéndice G	Unicode	Programming in C#	Deitel & Deitel	1349 - 1359

Se recomiendan estas lecturas para dominar el uso del estándar Unicode

27

JUSTIFICACIÓN

- Justificar a la izquierda con 10 espacios

```
- Console.WriteLine( "{ 0 , -10 }", x);
```

- Justificar a la derecha con 10 espacios

```
- Console.WriteLine( "{ 0 , 10 }", x);
```

28

COLOCAR EL CURSOR

- Para colocar el cursor en la pantalla utilice el método `SetCursorPosition()`

```
Console.SetCursorPosition(columna, renglon);
```

- Ejemplo

```
Console.SetCursorPosition(40, 12);  
Console.Write("Hola");
```

29

PROPIEDADES Y MÉTODOS DE LA CONSOLA

- Cambiar el color del fondo

```
Console.BackgroundColor = ConsoleColor.Red;  
Console.Clear();
```

- Cambiar el color del texto

```
Console.ForegroundColor = ConsoleColor.Yellow;  
Console.Write("Tec Laredo");
```

- Desaparecer el cursor

```
Console.CursorVisible = false;
```

30

PROPIEDADES Y MÉTODOS DE LA CONSOLA

- Cambiar la etiqueta de la ventana de la consola

```
Console.Title = "Tec Laredo";
```

- Emitir sonido

```
Console.Beep(frecuencia, duracion);
```

- Aparecer el cursor

```
Console.CursorVisible = true;
```

31

MÉTODOS Read()

- Para capturar datos desde el teclado:

```
string nombre;  
nombre=Console.ReadLine( );
```

```
int edad;  
edad = int.Parse(Console.ReadLine( ) );
```

```
float precio;  
precio = float.Parse(Console.ReadLine( ) );
```

32

COMENTARIOS

// Este es un comentario

/* Este es tambien un comentario
de varias lineas, que se utiliza
para parrafos
*/

33

```
using System;
using System.Collections.Generic;
using System.Text;

namespace Ejemplo
{
    class Program
    {
        static void Main(string[] args)
        {
            string nombre; // declara la variable nombre de tipo string (cadena)
            int edad; // declara la variable edad de tipo entero
            float estatura; // declara una variable estatura de tipo real

            Console.Clear(); // limpia la pantalla
            Console.WriteLine("Teclee su nombre: "); // despliega un mensaje en la pantalla
            nombre = Console.ReadLine(); // captura una cadena y la almacena en la variable nombre

            Console.WriteLine("\nTeclee la edad de {0}: ", nombre); // despliega un mensaje
            edad = int.Parse(Console.ReadLine()); // captura una cadena y la convierte a tipo entero

            Console.WriteLine("\nTeclee la estatura de {0}: ", nombre);
            estatura = float.Parse(Console.ReadLine()); // captura una cadena y la convierte a tipo real

            Console.WriteLine("\n\n{0} tiene {1} y mide {2}", nombre, edad, estatura);
            Console.WriteLine("\n\nOprima cualquier tecla para continuar ");
            Console.Read();
        }
    }
}
```

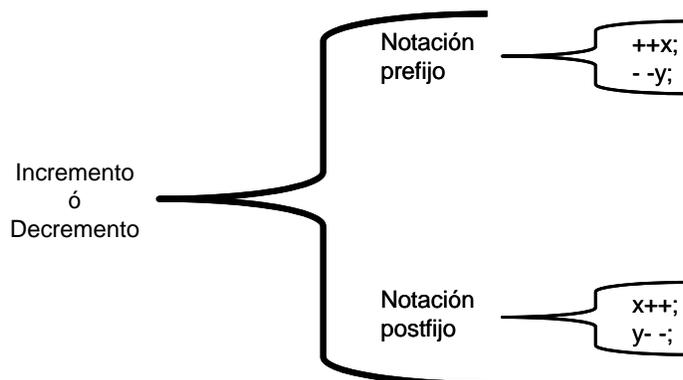
34

OPERADORES ARITMÉTICOS

Operador	Descripción	Ejemplo
+	Suma	res = num1 + num2;
-	Resta	res=num1 - num2;
*	Multiplicación	res = num1 * num2;
/	División entera	res = num1 / num2
/	División real	res = (float) num1 / (float) num2;
%	Residuo	res = num1 % num2;

35

INCREMENTOS Y DECREMENTOS



36