

Índice de contenidos

Introducción

Diez años de evolución tecnológica
Los desafíos del programador
Nace la era .NET
C#, un lenguaje actual
Objetivos de este libro

Cómo usar este libro

Estructura de la guía
Uso de los ejemplos

1. Instalación

1.1. Introducción
1.2. Obtener el paquete de desarrollo
1.3. Preparación del sistema
 1.3.1. Instalación del sistema operativo
 1.3.2. Internet Explorer 5.5
1.4. Instalar el SDK
1.5. Instalación de los ejemplos
1.6. En busca de un editor para C#

2. La primera aplicación

2.1. Introducción
2.2. Hola desde C#
 2.2.1. Metodología de trabajo
 2.2.2. Versión para consola
 2.2.3. Versión Windows
2.3. Elementos del lenguaje
 2.3.1. Clases y objetos
 2.3.2. El método Main()
 2.3.3. La consola y otros servicios
2.4. Proceso de compilación
 2.4.1. Una especificación común
 2.4.2. Obtener el código IL
2.5. Ejecución
 2.5.1. Funcionamiento del compilador JIT
2.6. Esquema general

3. Introducción a C#

3.1. Introducción
3.2. Unidades, módulos y aplicaciones
 3.2.1. Módulos de código
 3.2.2. Unidades lógicas
3.3. Espacios con nombre
 3.3.1. Definir espacios con nombre
 3.3.2. Referencias a un espacio con nombre
 3.3.3. Conflictos entre espacios con nombre
3.4. Clases
 3.4.1. Construcción y destrucción de objetos
 3.4.2. Constructores de clase
 3.4.3. Clases anidadas
 3.4.4. Campos de datos
 3.4.5. Visibilidad
 3.4.6. Métodos
 3.4.7. Propiedades
 3.4.8. Delegados y eventos
 3.4.9. Otros miembros de una clase

- 3.5. Interfaces
 - 3.5.1. Definir una interfaz
 - 3.5.2. Implementar una interfaz
- 3.6. Estructuras de control
- 3.7. Tipos y objetos
- 3.8. Resumiendo

4. Introducción a la plataforma Microsoft .NET

- 4.1. Introducción
- 4.2. Una red de servicios
- 4.3. El entorno común de ejecución
- 4.4. Código intermedio
- 4.5. Interoperabilidad entre lenguajes
- 4.6. Servicios .NET
- 4.7. Disponibilidad de la plataforma .NET
- 4.8. Proceso de estandarización
- 4.9. Términos .NET

5. Aplicaciones Windows

- 5.1. Introducción
- 5.2. Control de la aplicación
 - 5.2.1. Puesta en marcha de la aplicación
 - 5.2.2. Propiedades de la aplicación
 - 5.2.3. Otros aspectos de Application
- 5.3. Formularios
 - 5.3.1. Definición de una nueva clase de formulario
 - 5.3.2. Mostrar un formulario
 - 5.3.3. Insertar componentes
 - 5.3.4. Gestión de los eventos
- 5.4. Resumiendo

6. Aplicaciones Web

- 6.1. Introducción
- 6.2. De ASP a ASP+
 - 6.2.1. Libertad de elección en lenguajes
 - 6.2.2. Preparación del sistema
 - 6.2.3. Rendimiento de ASP+ frente a ASP
 - 6.2.4. Registro de componentes y bloqueo de librerías
- 6.3. Formularios web
 - 6.3.1. Uso de componentes C# desde ASP+
 - 6.3.2. Compilación dinámica
 - 6.3.3. Componentes de servidor
 - 6.3.4. Conexión entre diseño y lógica
 - 6.3.5. Validación de datos
- 6.4. Pagelets
 - 6.4.1. Creación de un pagelet
 - 6.4.2. Uso de un pagelet
- 6.5. Configuración
 - 6.5.1. Creación de un archivo config.web
 - 6.5.2. Cómo definir secciones propias
 - 6.5.3. Recuperar información de configuración
- 6.6. Resumiendo

7. Servicios Web

- 7.1. Introducción
- 7.2. ¿Qué es un servicio web?
 - 7.2.1. Servicios y aplicaciones
 - 7.2.2. Universalidad de un servicio
 - 7.2.3. Protocolos y lenguajes

- 7.3. Creación de un servicio web
 - 7.3.1. Anatomía de un servicio web
 - 7.3.2. Nuestro servicio horario
 - 7.3.3. Descripción SDL del servicio
- 7.4. Consumo del servicio
 - 7.4.1. Obtención de un proxy
 - 7.4.2. Un consumidor ASP+
 - 7.4.3. Un consumidor WinForms
- 7.5. Resumiendo

8. Desarrollo de componentes

- 8.1. Introducción
- 8.2. Conceptos y terminología
- 8.3. Componentes simples
 - 8.3.1. Un componente convertidor
 - 8.3.2. Uso desde una página ASP+
 - 8.3.3. Compilación del componente
 - 8.3.4. Uso en una herramienta de diseño
- 8.4. Conversión en forma de control
 - 8.4.1. Controles Windows
 - 8.4.2. Notificación de cambios en propiedades
 - 8.4.3. Controles Web
- 8.5. Controles compuestos
 - 8.5.1. Una ficha prefabricada
- 8.6. Mejorando el comportamiento de diseño
 - 8.6.1. Declaración de atributos
 - 8.6.2. Editores, convertidores y otros
- 8.7. Resumiendo

9. Acceso a datos

- 9.1. Introducción
- 9.2. El modelo ADO+
 - 9.2.1. Estructura de un DataSet
 - 9.2.2. Crear un DataSet a partir de una tabla
 - 9.2.3. Obtener la estructura de un DataSet
 - 9.2.4. Conexiones y comandos
 - 9.2.5. Acceso a los datos
- 9.3. Enlace con controles
 - 9.3.1. Enlace en páginas ASP+
 - 9.3.2. Enlace en aplicaciones Windows
- 9.4. ADO+ y Visual Studio.NET
- 9.5. Resumiendo

10. Interoperabilidad entre lenguajes

- 10.1. Introducción
- 10.2. ¿Qué es la CLS?
 - 10.2.1. Directrices de la CLS
 - 10.2.2. Cómo ajustarse a la CLS
- 10.3. Beneficios de la CLS
 - 10.3.1. Usar clases de otros lenguajes
 - 10.3.2. Excepciones entre lenguajes
 - 10.3.3. Herencia a través de lenguajes
 - 10.3.4. El secreto está en IL
- 10.4. Interoperabilidad con COM
 - 10.4.1. Usar componentes COM
 - 10.4.2. Exponer clases .NET como componentes COM
- 10.5. Resumiendo

11. Características avanzadas

- 11.1. Introducción
- 11.2. Múltiples hilos de ejecución
 - 11.2.1. Creación de un nuevo hilo de ejecución
 - 11.2.2. Problemas de sincronización
- 11.3. Código no seguro
- 11.4. Reflexión
 - 11.4.1. El tipo Type
 - 11.4.2. Cómo enumerar el contenido de una unidad
 - 11.4.3. Invocación dinámica
 - 11.4.4. Cómo ejecutar cualquier método con un parámetro entero

12. Visual Studio.NET

- 12.1. Introducción
- 12.2. Un entorno de desarrollo único
 - 12.2.1. La página de inicio
 - 12.2.2. Editores de Visual Studio.NET
 - 12.2.3. Creación de proyectos
 - 12.2.4. Otros elementos comunes
- 12.3. Herramientas para instalación
- 12.4. Resumiendo

A. Recursos en la red

- A.1. Introducción
- A.2. Lugares de referencia indispensable
 - A.2.1. Microsoft .NET
 - A.2.2. Microsoft .NET en MSDN
 - A.2.3. SOAP en DEVELOPMENTOR
 - A.2.4. World Wide Web Consortium
 - A.2.5. ECMA
 - A.2.6. XML
 - A.2.7. UDDI
 - A.2.8. Foros de conversación
- A.3. Artículos y código
 - A.3.1. Ejemplos y trucos sobre ASP+
 - A.3.2. El lenguaje C#
 - A.3.3. Otros recursos
- A.4. Recursos en castellano

Índice alfabético