

Programming in C#

20483

Detalles del Curso

Audiencia(s): Desarrolladores

Tecnología: Visual Studio

Duración: 40 Horas

Material Didáctico: Oficial (Ingles)

Sobre este Curso

Este curso de capacitación enseña a los desarrolladores las habilidades de programación que se requieren para que los desarrolladores creen aplicaciones de Windows utilizando el lenguaje Visual C#. Durante sus cinco días en el aula, los estudiantes revisan los conceptos básicos de la estructura del programa Visual C#, la sintaxis del lenguaje y los detalles de implementación, y luego consolidan sus conocimientos durante la semana a medida que crean una aplicación que incorpora varias características de .NET Framework 4.7.

Perfil de Audiencia

Este curso está dirigido a desarrolladores experimentados que ya tienen experiencia en programación en C, C ++, JavaScript, Objective-C, Microsoft Visual Basic o Java y entienden los conceptos de programación orientada a objetos. Este curso no está diseñado para estudiantes que son nuevos en la programación. Está dirigido a desarrolladores profesionales con al menos un mes de experiencia en programación en un entorno orientado a objetos.

Contenido del Curso

Módulo 1: Revisión de la sintaxis de Visual C#

Microsoft .NET Framework versión 4.7 proporciona una plataforma de desarrollo integral que puede usar para construir, implementar y administrar aplicaciones y servicios. Al usar .NET Framework, puede crear aplicaciones visualmente atractivas, permitir una comunicación fluida a través de los límites de la tecnología y brindar soporte para una amplia gama de procesos de negocios.

En este módulo, aprenderá sobre algunas de las características principales proporcionadas por .NET Framework y Microsoft Visual Studio. También aprenderá sobre algunas de las construcciones principales de Visual C# que le permiten comenzar a desarrollar aplicaciones .NET Framework.

Lecciones

- Descripción general de la aplicación de escritura mediante Visual C#
- Tipos de datos, operadores y expresiones
- Construcciones del lenguaje de programación Visual C#

Laboratorio: Implementación de la funcionalidad editar para la lista de estudiantes

Módulo 2: Creación de métodos, manejo de excepciones y aplicaciones de supervisión

Las aplicaciones generalmente consisten en unidades lógicas de funcionalidad que realizan funciones específicas, como proporcionar acceso a datos o activar algún procesamiento lógico. Visual C# es un lenguaje orientado a objetos y utiliza el concepto de métodos para encapsular unidades lógicas de funcionalidad. Un método puede ser tan simple o complejo como desee y, por lo tanto, es importante tener en cuenta qué sucede con el estado de su aplicación cuando se produce una excepción en un método.

En este módulo, aprenderá cómo crear y usar métodos y cómo manejar excepciones. También aprenderá a usar el registro y el seguimiento para registrar los detalles de cualquier excepción que ocurra.

Lecciones

- Crear e invocar métodos
- Creación de métodos sobrecargados y uso de parámetros opcionales y de salida
- Manejo de excepciones
- Supervisión de aplicaciones

Laboratorio: Ampliación de la funcionalidad de la aplicación de inscripción de clase

Módulo 3: Tipos básicos y construcciones de Visual C#

Para crear aplicaciones efectivas utilizando Windows Presentation Foundation (WPF) u otras plataformas .NET Framework, primero debe aprender algunas construcciones básicas de Visual C#. Necesita saber cómo crear estructuras simples para representar los elementos de datos con los que está trabajando. Debe saber cómo organizar estas estructuras en colecciones para poder agregar elementos, recuperar elementos e iterar sobre ellos. Finalmente, necesita saber cómo suscribirse a eventos para poder responder a las acciones de sus usuarios.

En este módulo, aprenderá a crear y usar estructuras y enumeraciones, organizar datos en colecciones y crear y suscribirse a eventos.

Lecciones

- Implementación de estructuras y enumeraciones
- Organización de datos en colecciones
- Manejo de eventos

Laboratorio: Escribir el código para la aplicación del prototipo de grados

Módulo 4: Crear clases e implementar colecciones de tipos seguros

En este módulo, aprenderá a usar interfaces y clases para definir y crear sus propios tipos personalizados y reutilizables. También aprenderá cómo crear y usar colecciones de cualquier tipo enumerables y seguras.

Lecciones

- Crear clases
- Definición e implementación de interfaces
- Implementación de colecciones de tipo seguro

Laboratorio: Agregar validación de datos y seguridad de tipo a la aplicación

Módulo 5: Crear una jerarquía de clases mediante el uso de herencia

En este módulo, aprenderá a usar la herencia para crear jerarquías de clases y extender los tipos de .NET Framework.

Lecciones

- Crear jerarquías de clases
- Extender las clases de .NET Framework

Laboratorio: Refactorizar la funcionalidad común en la clase de usuario

Módulo 6: Leer y escribir datos locales

En este módulo, aprenderá a leer y escribir datos utilizando operaciones de E/S del sistema de archivos transaccionales, cómo serializar y deserializar datos al sistema de archivos, y cómo leer y escribir datos en el sistema de archivos mediante el uso de flujos.

Lecciones

- Lectura y escritura de archivos
- Serialización y deserialización de datos
- Realización de E/S mediante el uso de secuencias

Laboratorio: Generar el informe de calificaciones

Módulo 7: Acceder a base de datos

En este módulo, aprenderá cómo crear y usar modelos de datos de entidad (EDM) y cómo consultar muchos tipos de datos mediante la consulta integrada de lenguaje (LINQ).

Lecciones

- Creación y uso de modelos de datos de entidad
- Consulta de datos mediante LINQ

Laboratorio: Recuperar y modificar datos de calificaciones

Módulo 8: Acceder a datos remotos

En este módulo, aprenderá a usar las clases de solicitud y respuesta en el espacio de nombres System.Net para manipular directamente las fuentes de datos remotas. También aprenderá a usar los Servicios de datos de Windows Communication Foundation (WCF) para exponer y consumir un modelo de datos de entidad (EDM) en la web.

Lecciones

- Acceso a datos a través de la web
- Acceso a datos mediante el uso de servicios conectados de ODa

Laboratorio: Recuperar y modificar datos de calificaciones de manera remota

Módulo 9: Diseño de la interfaz de usuario para una aplicación gráfica

En este módulo, aprenderá a usar el Lenguaje de marcado de aplicación extensible (XAML) y Windows Presentation Foundation (WPF) para crear interfaces de usuario atractivas.

Lecciones

- Uso de XAML para diseñar una interfaz de usuario
- Controles vinculantes a datos

Laboratorio: Personalización de fotografías de estudiantes y diseño de la aplicación

Módulo 10: Mejorar el rendimiento de la aplicación y la capacidad de respuesta

En este módulo, aprenderá cómo mejorar el rendimiento de sus aplicaciones distribuyendo sus operaciones en varios subprocesos.

Lecciones

- Implementación de tareas múltiples
- Realización de operaciones asincrónicas
- Sincronización de acceso concurrente a datos

Laboratorio: Mejorar la capacidad de respuesta y el rendimiento de la aplicación

Módulo 11: Integración con código no administrado

En este módulo, aprenderá cómo interoperar código no administrado en sus aplicaciones y cómo asegurarse de que su código libere recursos no administrados.

Lecciones

- Crear y usar objetos dinámicos
- Administrar la vida útil de los objetos y controlar los recursos no administrados

Laboratorio: Actualización el informe de calificaciones

Módulo 12: Crear tipos y ensamblados reutilizables

En este módulo, aprenderá cómo consumir ensamblados existentes mediante el uso de la reflexión y cómo agregar metadatos adicionales a los tipos y miembros de tipo mediante el uso de atributos. También aprenderá cómo generar código en tiempo de ejecución utilizando el Modelo de objeto de documento de código (CodeDOM) y cómo asegurarse de que sus ensamblajes estén firmados y versionados, y disponibles para otras aplicaciones, utilizando el caché de ensamblados global (GAC).

Lecciones

- Examen de metadatos de objetos
- Crear y usar atributos personalizados
- Generar código administrado
- Versiones, firma y despliegue de ensamblados

Laboratorio: Especificar los datos a incluir en el informe de calificaciones

Módulo 13: Cifrar y descifrar datos

En este módulo, aprenderá cómo implementar el cifrado simétrico y asimétrico y cómo usar hashes para generar representaciones matemáticas de sus datos. También aprenderá cómo crear y administrar certificados X509 y cómo usarlos en el proceso de cifrado asimétrico.

Lecciones

- Implementación de cifrado simétrico
- Implementación de cifrado asimétrico

Laboratorio: Cifrar y descifrar el informe de calificaciones

Al Finalizar este Curso

Después de completar este curso, los estudiantes podrán:

- Describir la sintaxis central y las características de Visual C#.
- Crear métodos, manejar excepciones y describir los requisitos de supervisión de aplicaciones a gran escala.
- Implementar la estructura básica y los elementos esenciales de una aplicación de escritorio típica.
- Crear clases, defina e implemente interfaces, y crear y usar colecciones genéricas.
- Usar la herencia para crear una jerarquía de clases y extender una clase de .NET Framework.
- Leer y escribir datos utilizando la entrada/salida de archivos y flujos, y serializar y deserializar datos en diferentes formatos.
- Crear y usar un modelo de datos de entidad para acceder a una base de datos y usar LINQ para consultar datos.
- Acceder y consultar datos remotos utilizando los tipos en el espacio de nombres System.Net y WCF Data Services.
- Crear una interfaz gráfica de usuario utilizando XAML.
- Mejorar el rendimiento y el tiempo de respuesta de las aplicaciones mediante el uso de tareas y operaciones asíncronas.
- Integrar bibliotecas no administradas y componentes dinámicos en una aplicación de Visual C#.
- Examinar los metadatos de los tipos mediante la reflexión, cree y usar atributos personalizados, generar código en tiempo de ejecución y administrar versiones de ensamblaje.
- Cifrar y descifrar datos mediante el cifrado simétrico y asimétrico.