www.logicalimagination.com
800.657.1494

.NET Framework Using C#

Course #: VC-710 **Duration:** 4 days

Prerequisites

The student should be an experienced application developer or architect with a working knowledge of C#, including building simple GUIs with Windows Forms.

Details

This four-day course is designed to provide a sound introduction to the .NET Framework for programmers who already know the C# language and the fundamentals of Windows Forms. It is current to Visual Studio 2019 or higher, including support for cross-platform development using .NET Core. The course focuses on core portions of the .NET Framework that are common across many application areas. Separate courses are available in specific areas, such as ADO.NET, XML Programming, Windows Presentation Framework, Windows Communications Framework, and ASP.NET.

The course starts with an introduction to the architecture and key concepts of .NET. It then discusses class libraries, assemblies, versioning, configuration, and deployment, which constitute a significant advance in the simplicity and robustness of deploying Windows applications, ending the notorious "DLL hell." You learn important topics in the .NET programming model, including metadata, reflection, I/O, and serialization, and the .NET programming model, covering memory management, asynchronous programming, and application domains. Finally, you learn about threading, which includes an introduction to the Task Parallel Library (TPL).

.NET Security, which was simplified in .NET 4.0, is introduced, including both code access security and role-based security. The interoperability of .NET with COM and with Win32 applications is discussed as well as an introduction to database programming using ADO.NET and LINQ. Finally, the .NET Framework diagnostic facilities are discussed in depth.

The course is practical, with many examples and a case study. The goal is to equip you to begin building significant applications using the .NET Framework.

Software Needed

The required software is Visual Studio 2019 or newer. The free Visual Studio Community 2019 or higher may be used.

Outline

.NET Framework Using C#

- **.NET Fundamentals**
  - What is Microsoft .NET?
  - Common Language Runtime
  - CLR Serialization
  - Attribute-Based Programming
  - Interface-Based Programming
  - Metadata
  - Common Type System

    - Framework Class Library
    - Language Interoperability
    - Managed Code
    - Assemblies and Deployment
    - Web Services
    - ASP.NET
    - Performance
    - .NET Native
    - .NET Core and Cross-platform Development
    - XML Serialization
- **Class Libraries**
    - Components in .NET
    - Building Class Libraries at the Command Line
    - Class Libraries Using Visual Studio
    - Using References
- **Assemblies, Deployment and Configuration**
    - Assemblies
    - Private Assembly Deployment
    - Shared Assembly Deployment
    - Configuration Overview
    - Configuration Files
    - Programmatic Access to Configuration
    - Using SDK Tools for Signing and Deployment
    - Application Settings
- **Metadata and Reflection**
    - Metadata
    - Reflection
    - Late Binding
- **I/O and Serialization**
    - Directories
    - Files
    - Serialization
    - Attributes
- **.NET Programming Model**
    - Memory Management and Garbage Collection
    - Asynchronous Delegates
    - BackgroundWorker
    - Application Domains
- **.NET Threading**
    - Threading Fundamentals
    - ThreadPool
    - Foreground and Background Threads
    - Synchronization
    - Task Parallel Library
- **.NET Security**
    - Authentication and Authorization
    - Code Access Security
    - Sandboxing
    - Permissions
    - Role-Based Security
    - Principals and Identities
- **Interoperating with COM and Win32**
    - .NET Client Calling a COM Server
    - 64-bit System Considerations
    - PInvoke
- **.NET and LINQ**
    - ADO.NET Overview
    - .NET Data Providers
    - Connections
    - Using LocalDB
    - Commands
    - DataReaders and Connected Access
    - Data Sets and Disconnected Access
    - Language Integrated Query
- **Debugging Fundamentals**

- Compile-time Errors and Run-time Errors
- Configuring Debug, Release, and Special Builds
- Visual Studio Debugger
- Just-In-Time Debugging
- Attaching Debugger to a Running Process

- **Tracing**
  - Tracing
  - Event Logs
  - Using the BooleanSwitch and TraceSwitch Classes
  - Print Debugging Information with the Debug Class
  - Instrumenting Release Builds with the Trace Class
  - Using Listeners
  - Implementing Custom Listeners