

C# and .NET Foundations (4014)

Duration: 5 days (40 hours)

This five-day course is designed for the experienced programmer to help you quickly come up to speed on the essentials of the C# language and the .NET Core frameworks. It is current to Visual Studio 2022, .NET 6 and C# 10.

The first module of the course covers the C# language and core features of C# and .NET, including delegates and events and an introduction to Windows Forms. The second module focuses on core portions of the .NET Class Library that are common across many application areas. It starts with an introduction to the architecture and key concepts of .NET. The course then discusses class libraries, packages, metapackages and frameworks. The following chapters discuss important topics in the .NET programming model, including metadata and reflection, I/O and serialization, memory management, processes and threads. The course concludes with a chapter on threading, which includes an introduction to the Task Parallel Library (TPL).

The course is hands-on, with many programming examples. The goal is to equip you with the foundations of this important technology from Microsoft. The student will receive a comprehensive set of materials, including course notes and all the programming examples.

Benefits

- Acquire a working knowledge of C# programming
- Learn about important interactions between C# and .NET
- Learn how to implement simple GUI programs using Windows Forms
- Gain a thorough understanding of the philosophy and architecture of .NET Core
- Understand packages, metapackages and frameworks
- Acquire a working knowledge of the .NET programming model
- Implement multi-threading effectively in .NET applications

Outline

Module 1: C# Essentials

1. Introduction to NET

- What is .NET?
- .NET Framework, NET Core and .NET 6
- Application Models
- Managed Code
- Visual Studio 2022
- Console Programs and New Console Template
- GUI Programs

2. C# Overview for the Sophisticated Programmer

- First C# Console Application
- Namespaces
- Data Types
- Conversions
- Control Structures

- Subroutines and Functions
- Parameter Passing
- Strings
- Arrays
- Implicitly Typed Variables
- Console I/O
- Formatting
- Exception Handling

3. Object-Oriented Programming in C#

- Classes
- Access Control
- Methods and Properties
- Asymmetric Accessor Accessibility
- Static Data and Methods
- Constant and Readonly Fields
- Auto-Implemented Properties
- Inheritance
- Overriding Methods
- Abstract Classes
- Sealed Classes
- Access Control and Assemblies

4. C# and .NET

- Components
- Interfaces
- System.Object
- Collections
- IEnumerable and IEnumerator
- Copy Semantics in C#
- Generic Types
- Type-Safe Collections
- Object Initializers
- Collection Initializers
- Anonymous Types
- Attributes

5. Delegates and Events

- Delegates
- Anonymous Methods
- Lambda Expressions
- Random Number Generation
- Events

6. Introduction to Windows Forms

- Creating Windows Applications Using Visual Studio 2022
- Partial Classes
- Buttons, Labels and Textboxes
- Handling Events

- Listbox Controls

7. Newer Features in C#

- Dynamic Data Type
- Named and Optional Arguments
- Variance in Generic Interfaces
- Asynchronous Programming Keywords
- New Features in C# 6 and C# 7
- Nullable Reference Types
- Record Types
- Top-level Statements

Appendix A. Using Visual Studio 2022

- Signing in to Visual Studio
- Overview of Visual Studio 2022
- Creating a Console Application
- Project Configurations
- Debugging
- Multiple-Project Solutions

Appendix B. Language Integrated Query (LINQ)

- What Is LINQ?
- Basic Query Operators
- Filtering
- Ordering
- Aggregation

Appendix C. Unsafe Code and Pointers in C#

- Unsafe Code
- C# Pointer Type

Module 2: .NET Foundations

1. NET Fundamentals

- What is Microsoft .NET?
- Common Language Runtime
- Framework Class Library
- Language Interoperability
- Managed Code
- .NET Core, .NET 7, and Cross-Platform Development

2. Class Libraries

- Components in .NET
- Class Libraries Using Visual Studio
- Using References

3. Frameworks and Packages

- .NET Implementations and Frameworks
- .NET Standard
- NuGet Packages and Gallery
- Metapackages and Frameworks
- Packages in .NET 7
- Porting from Classical .NET to .NET 7
- Visual Studio Package Manager
- Installing Packages
- Creating Packages

4. Metadata and Reflection

- Metadata
- Reflection
- Late Binding

5. I/O and Serialization

- Directories
- Files and Streams
- XML Serialization

6. .NET Programming Model

- Garbage Collection
- Finalize and Dispose
- Processes
- Command-Line Arguments
- Threads

7. .NET Threading

- Threading Fundamentals
- ThreadPool
- Foreground and Background Threads
- Synchronization
- Task Parallel Library

Required Prerequisites

The student should be an experienced application developer or architect.

Useful Prerequisites

Some background in object-oriented programming in a language such as C++ or Java would be helpful.