

Course duration

- 5 days

Course Benefits

- Acquire a working knowledge of C# programming
- Learn how to implement programs using C# and classes from the .NET Framework
- Learn how to implement simple GUI programs using Windows Forms
- Gain a working knowledge of important newer features in C#

Available Delivery Methods

Public Class

Public expert-led online training from the convenience of your home, office or anywhere with an internet connection. Guaranteed to run .

Private Class

Private classes are delivered for groups at your offices or a location of your choice.

Course Outline

1. Introduction to NET
 1. What is .NET?
 2. .NET Framework and .NET Core
 3. Application Models
 4. Managed Code
 5. Visual Studio 2019
 6. C# Console and GUI Programs
2. First C# Programs
 1. Hello, World
 2. Namespaces
 3. Variables and Expressions
 4. Using C# as a Calculator
 5. Input/Output in C#
 6. .NET Framework Class Library
3. Data Types in C#
 1. Data Types
 2. Integer Types

- 3. Floating Point Types
- 4. Decimal Type
- 5. Characters and Strings
- 6. Boolean Type
- 7. Conversions
- 8. Nullable Types
- 4. Operators and Expressions
 - 1. Operator Cardinality
 - 2. Arithmetic Operators
 - 3. Relational Operators
 - 4. Logical Operators
 - 5. Bitwise Operators
 - 6. Assignment Operators
 - 7. Expressions
 - 8. Checked and Unchecked
- 5. Control Structures
 - 1. If Tests
 - 2. Loops
 - 3. Arrays
 - 4. Foreach
 - 5. More about Control Flow
 - 6. Switch
- 6. Object-Oriented Programming
 - 1. Objects
 - 2. Classes
 - 3. Inheritance
 - 4. Polymorphism
 - 5. Object-Oriented Languages
 - 6. Components
- 7. Classes
 - 1. Classes as Structured Data
 - 2. Methods
 - 3. Constructors and Initialization
 - 4. Static Fields and Methods
 - 5. Constant and Readonly
- 8. More about Types
 - 1. Overview of Types in C#
 - 2. Value Types
 - 3. Boxing and Unboxing
 - 4. Reference Types
 - 5. Implicitly Typed Variables
- 9. Methods, Properties and Operators
 - 1. Methods
 - 2. Parameter Passing
 - 3. Method Overloading
 - 4. Variable-Length Parameter Lists
 - 5. Properties

- 6. Auto-Implemented Properties
- 7. Operator Overloading
- 10. Characters and Strings
 - 1. Characters
 - 2. Strings
 - 3. String Input
 - 4. String Methods
 - 5. StringBuilder Class
 - 6. Programming with Strings
- 11. Arrays and Indexers
 - 1. Arrays
 - 2. System.Array
 - 3. Random Number Generation
 - 4. Jagged Arrays
 - 5. Rectangular Arrays
 - 6. Arrays as Collections
 - 7. Bank Case Study—Step 1
 - 8. Indexers
- 12. Inheritance
 - 1. Single Inheritance
 - 2. Access Control
 - 3. Method Hiding
 - 4. Initialization
 - 5. Bank Case Study—Step 2
- 13. Virtual Methods and Polymorphism
 - 1. Virtual Methods and Dynamic Binding
 - 2. Method Overriding
 - 3. Fragile Base Class Problem
 - 4. Polymorphism
 - 5. Abstract Classes
 - 6. Sealed Classes
 - 7. Heterogeneous Collections
 - 8. Bank Case Study—Step 3
- 14. Formatting and Conversion
 - 1. ToString
 - 2. Format Strings
 - 3. String Formatting Methods
 - 4. Bank Case Study—Step 4
 - 5. Type Conversions
- 15. Exceptions
 - 1. Exception Fundamentals
 - 2. Structured Exception Handling
 - 3. User-Defined Exception Classes
 - 4. Inner Exceptions
 - 5. Bank Case Study—Step 5
- 16. Interfaces
 - 1. Interface Fundamentals

2. Programming with Interfaces
3. Using Interfaces at Runtime
4. Bank Case Study—Step 6
5. Resolving Ambiguities
17. .NET Interfaces and Collections
 1. Collections
 2. Bank Case Study—Step 7
 3. IEnumerable and IEnumerator
 4. Copy Semantics and ICloneable
 5. Comparing Objects
 6. Generic Types
 7. Type-Safe Collections
 8. Object Initializers
 9. Collection Initializers
 10. Anonymous Types
 11. Bank Case Study—Step 8
18. Delegates and Events
 1. Delegates
 2. Anonymous Methods
 3. Lambda Expressions
 4. Events
19. Introduction to Windows Forms
 1. Creating Windows Applications Using Visual Studio 2019
 2. Partial Classes
 3. Buttons, Labels and Textboxes
 4. Handling Events
 5. Listbox Controls
20. Newer Features in C#

Class Materials

Each student will receive a comprehensive set of materials, including course notes and all the class examples.

Class Prerequisites

Experience in the following *is required* for this C# class:

- Programming experience in a high-level language.