

The C# Programming

Learn via: **Classroom / Virtual Classroom / Online**

Duration: **5 Day**

Overview

Microsoft's .NET Framework presents developers with unprecedented opportunities. From web applications to desktop and mobile platform applications - all can be built with equal ease, using substantially the same skill-set. But to make the most of this potential, developers must have a thorough grasp of core language skills and OO programming concepts.

This {training} concentrates on the C# programming language itself, to prepare delegates fully in readiness for exploring the

.NET Framework. No Object-Oriented knowledge is assumed - the course provides a suitable OO primer. From basic procedural syntax to sophisticated object-oriented programming techniques, delegates will learn how to write .NET applications with code that is robust and maintainable.

The {training} is presented as a mixture of lectures, demos and hands-on exercises. Practical sessions follow all main topics, designed to reinforce the points covered. Additional information is provided in appendices to extend the learning experience after the course has been completed.

The course provides a starting point for delegates wishing to undertake Microsoft exam 70-483 - Programming in C# although further study is likely to be required before sitting the exam.

Who Should Attend

This intensive course is intended for developers who will use C# to write .NET Framework applications and who are relatively new to the C# programming language.

Delegates looking to migrate their existing Java and/or C skills to C# should attend our QAJAVCSMIG - Migrating to C# for Java and C developers.

What You Will Learn

At the end of this {training}, attendees will have an understanding of the following:

- Write code that includes sequence, selection and iteration constructs
- Create and use classes and structures (types), including fields, properties and methods
- Use private, internal, protected and public visibility modifiers
- Use exception-handling to create robust applications
- Create derived classes that inherit from custom-written or .NET Framework classes
- Create interfaces and apply techniques of polymorphism effectively and appropriately
- Work with generic types
- Leverage the power of C# features
- Use delegates
- Understand the event handling paradigm
- Use lambda expressions in C# code
- Use LINQ (Language Integrated Query)
- Efficiently manage resources

Outline

Module 1: Introduction to Object Orientation

- This ensures all delegates have a grounding in the principles of Object Orientation.

Module 2: OO and C Sharp

- Get started on how C# implements Object Orientation

Module 3: Introduction to .NET Visual Studio & C#

- The .NET Framework; The Common Language Runtime; The Common Type System
- C# Features; Introduction to namespaces and assemblies
- Get to know your way around Visual Studio

Module 4: Syntax I

- Procedures and statements; Data types; Declaring variables; Assignments
- Conversion; Arithmetic and other operators
- Flow of control constructs
- Passing parameters by value, by reference, named and optional parameters

Module 5: Syntax 2

- Type concepts; Classes; Reference types
- Fields, properties and methods
- Accessibility modifiers
- Object initialisation, Constructors and Constructor chaining
- Instance members; Keyword 'this'
- The 'null' reference
- const & readonly
- Enumerated types
- Arrays

Module 6: Collections

- Generic Collections
- Iterating collections
- Indexers

Module 7: Inheritance & Polymorphism

- Concept of inheritance; Substitutability; Extending a simple class
- 'virtual', 'override' and 'sealed' modifiers
- Polymorphism
- Abstract classes
- Upcasting and safe downcasting

Module 8: Interfaces

- Polymorphism with interfaces
- Multiple interfaces

Module 9: Delegates and Lambdas

- Delegates explained
- Working with delegates
- Creating your own delegate types
- Lambdas

Module 10: Generic Delegates

- Using the framework-supplied generic delegates

Module 11: LINQ

- The language features behind LINQ

- The LINQ API and query pattern
- Grouping and Joins

Module 12: Handling Exceptions

- Errors vs. Exceptions
- The 'try', 'catch', 'finally' paradigm
- Using 'throw'
- Creating your own exceptions
- 'checked' and 'unchecked' expressions.

Module 13: Consuming Events

- Understanding events in .NET
- Writing event handlers

Module 14: Producing Events

- The event conventions
- Raising custom events

Module 15: Garbage Collection

- Garbage collection and its impact
- Finalizers; The 'Dispose' pattern; IDisposable
- The 'using' statement

Module 16: Reverse Engineer

- At the end of the course we reverse-engineer the resultant code and compare the results to the Class Diagram we produced near the beginning of the course.

Module 17: The Way Ahead

- Review

Appendix Module A: LINQ to Xml

- Producing and consuming XML using LINQ