

## Course 461: C# 2.0 Programming with Visual Studio .NET 2005 (5 days)

### Course Description...

The release of .NET 1.0 provided developers with a powerful object-oriented development environment that could be used to create client-server and web-based applications. These applications were capable of leveraging different data access technologies, XML, Web Services, and much more using a variety of languages including C#. With the release of .NET 2.0, the Visual Studio .NET and the C# language have been significantly upgraded to enable developers to build applications more quickly than ever. The *C# 2.0 Programming with Visual Studio .NET 2005* course provides an in-depth, hands-on analysis of the C# language and demonstrates how to develop flexible .NET applications using Visual Studio .NET 2005.

The course begins by providing a look at the different components of the .NET framework and discusses several differences between Java, C++, and C#. Next, the basic building blocks of the C# language including variables, conditionals and looping are discussed followed by details on how to use fields, properties, methods, classes, and namespaces.

Object-oriented features found in C# are also covered so that students understand how to leverage concepts such as inheritance, method overriding, interfaces, and object hierarchies. After covering the fundamentals of C#, the course shows one of the more exciting new language features in .NET version 2: Generics. Other topics covered include ADO.NET data access technologies, threading, deployment as well as Windows Forms and ASP.NET Web Forms.

This course is written by Microsoft .NET MVP Dan Wahlin.

### Learning Objectives...

- C# Language Fundamentals
- Object-oriented principles
- C# Generics and other C# 2.0 features
- Visual Studio .NET 2005 features
- The role interfaces play in .NET
- The role of classes, objects, namespaces, and assemblies



- .NET Base Class Libraries
- Components of the .NET framework
- Windows Forms Fundamentals
- New features in ASP.NET 2.0 Web Forms
- ADO.NET and data binding
- Working with Multiple Threads
- ClickOnce deployment techniques

## **Who should attend...**

This course is valuable for developers and engineers who are interested in programming using C#.

## **Prerequisites...**

Previous experience with a programming language such as Visual Basic, C++, or Java is highly recommended to get the most out of this course.

**See next page for a detailed course outline...**



## Course Outline...

### Section 1 – Building Applications with the .NET Framework

Before any .NET language is learned students must understand the different components found in the .NET framework. In this section of the course students will be provided with details about the different components.

- What is the .NET Framework?
- Working with Managed Code
  - .NET object-oriented language choices
  - Multi-language interoperability
- The role of the Common Language Runtime (CLR)
  - What is the CLR and what does it do in .NET?
  - Memory Management and Garbage Collection
  - What is the Common Type System (CTS)?
  - Microsoft Intermediate Language (MSIL)
  - Just-in-Time compilation
  - Versioning support

### Section 2 –C# Fundamentals

- What applications can you build with C#?
- C# Language Features in C# 2.0
  - Compiled
  - Object-oriented
  - Threading support
  - Structured exception handling
  - Garbage collection
- New Features in C# 2.0
  - Generics
  - Anonymous Methods
  - Yield keyword and collections
  - Partial Types
- Comparing C# to Java and C++
- Creating a C# Hello World Application
- Compiling C# applications
- **Hands-on lab:** Creating your first C# Program

### Section 3 – Introduction to Visual Studio.NET 2005

- Why use VS.NET 2005?
- Understanding Solutions and Projects
- Creating a C# solution and project using VS.NET 2005
  - Project Configuration
  - Compiling projects
  - Debugging projects



- Using VS.NET 2005 features
  - Class Designer
  - Code Snippets
  - Refactoring Support
  - Debugger Visualizers
- **Hands-on lab:** Creating VS.NET 2005 Projects

## Section 4 – C# Language Fundamentals

- C# Basics
  - Writing statements
  - Commenting Code
  - XML Comments
  - Value vs. Reference Types
  - C# Operators
- Defining Variables, Constants, and Arrays in C#
  - What is a variable?
  - Understanding variable scoping
  - Declaring variables
  - Declaring constants
  - Declaring arrays
- Conditionals and Looping
  - if..else statements
  - ? and : operators
  - switch statements
  - looping statements

## Section 5 – Building Classes and Structs

- What are classes?
- What are objects?
- Class members
- Creating a class from scratch
- Creating a class in VS.NET 2005 with the Class Designer
- Setting class member protection levels
- Adding constructors
- Adding fields to a class
- Adding property set and get statements to a class
- Property indexers
- Adding methods to a class
  - Defining method return types
  - Method parameters
  - Returning data from a method
    - Returning by value
    - Returning by reference
  - Anonymous Methods



- What is a struct?
- Differences between classes and structs
- The role of Namespaces and assemblies
  - Resolving naming conflicts with Namespaces
  - What is an assembly?
  - Private Vs. Shared Assemblies
  - The Global Assembly Cache (GAC)
- **Hands-on lab:** Creating a C# class with members  
Creating a C# class using the VS.NET 2005 Class Designer

## Section 6 – Object-Oriented Fundamentals

- What is object-oriented programming?
- The role of System.Object in .NET
- Understanding Abstraction, Encapsulation, Polymorphism and Inheritance
- Using abstract classes
- Working with interfaces
  - Defining interfaces
  - Implementing interfaces
- Understanding boxing and unboxing in .NET
- Operator Overloading
- Using .NET Attributes
- Structured Exception handling
  - Error handling in C#
  - Exception objects
  - Using Try...Catch blocks
  - Adding a Finally block
  - Throwing exceptions
- **Hands-on lab:** Inheriting from a base class

## Section 7 – Generics

- What are Generics?
- System.Collections.Generic Classes
- Creating objects using Generics
- Defining a custom Generic type
- Generic Constraints
- Using the Nullable Struct
- The default keyword
- **Hands-on lab:** Using Generics in Classes

## Section 8 – Delegates and Events

- The role of Events and Delegates in .NET
- Understanding events in C# applications



- Understanding and Creating Events
- Understanding and Creating Delegates
- Hooking up Event Handlers
- Using Anonymous Methods with Events
- **Hands-on lab:** Adding Delegates and Events to a class

## Section 9 – Using the .NET Framework Class Library

- Overview of functionality in the framework class library
- Working with System.IO classes
  - Stream readers and writers
  - Reading files
  - Writing to files
  - Working with Memory Streams
- Building strings with the StringBuilder class
- Working with Collections
- Working with Dates and Times
- Accessing remote data and sending email with System.Net classes
- Pattern searching with Regular Expressions
- **Hands-on lab:** Reading and writing to the file system

## Section 10 – Creating Windows Forms Applications with C#

- What are Windows Forms?
- Windows Forms Features
- Classes in the System.Windows.Forms namespace
- Creating a Windows Forms form
- Adding Windows Controls to forms
  - Adding menus
  - Adding input controls
- Handling Events
- Drawing and Graphics with GDI+
- Visual Inheritance
- Application Deployment Options
  - ClickOnce Deployment
- **Hands-on lab** – Creating a Windows Form application in Visual Studio.NET 2005

## Section 11 – Data Access with ADO.NET

- What is ADO.NET?
- What's new in ADO.NET 2.0?
- Managed Provider Classes in ADO.NET
- Connecting to a data base



- Using the Command Object
  - Calling stored procedures
  - Passing parameters to stored procedures
  - Accessing output parameters
- Using the DataReader Class
- Using the DataAdapter and DataSet classes
- Generic database access with DbProviderFactory classes
- Binding data to Windows Forms Controls
- **Hands-on lab** – Adding ADO.NET functionality to C# applications

## Section 12 – Creating ASP.NET Web Forms with C#

- What are Web Forms?
- New Features in ASP.NET 2.0
- Using Server Controls in Web Forms
  - What are Server Controls?
  - What types of server controls are there?
  - Programming server controls
  - Handling events
- Binding data using data source controls
  - ObjectDataSource
  - SqlDataSource
  - XmlDataSource
- **Hands-on lab** - Creating a Web Form application in Visual Studio.NET 2005

## Section 13 – Advanced C# Features

- Working with Threads
  - SyncLock
  - Good and bad threading techniques
  - Using the BackgroundWorker component
- Versioning
  - Signing assemblies
  - Adding assemblies to the GAC
  - Using versioned components
- XML Serialization
  - Creating XSD schemas
  - Using xsd.exe to generate code from schemas
  - Using the XmlSerializer Class
- Dynamically invoking objects with Reflection
- Integrating COM objects into .NET applications with COM Interop

***Please contact your ROI representative to discuss course tailoring!!!***