

## MS 2124C - Programming with C#

**Description:** The goal of this course is to provide students with the knowledge and skills they need to develop C# applications for the Microsoft .NET Platform. The course focuses on C# program structure, language syntax, and implementation details. C# was created to be the programming language best suited for writing enterprise applications for .NET. C# combines the high productivity of Microsoft Visual Basic with the raw power of C++. It is a simple, object-oriented, and type-safe programming language that is based on the C and C++ family of languages.

**Days:** 5

**Prerequisites:** Before attending this workshop, students must:

- Experience with programming in C, C++, Visual Basic, Java, or another programming language.
- Familiarity with the Microsoft .NET strategy as described on the Microsoft .NET Web site:  
<http://www.microsoft.com/net/>
- Familiarity with the .NET Framework as described on the MSDN Magazine Web site:  
<http://msdn.microsoft.com/msdnmag/issues/0900/Framework/Framework.asp>  
<http://msdn.microsoft.com/msdnmag/issues/1000/Framework2/Framework2.asp>

**Audience:** This course is intended for experienced developers who already have programming experience in C, C++, Visual Basic, or Java. These developers will be likely to develop enterprise business solutions.

### Module 1: Overview of the Microsoft .NET Platform

- Introduction to the .NET Platform
- Overview of the .NET Framework
- Benefits of the .NET Framework
- The .NET Framework Components
- Languages in the .NET Framework

## **Module 2: Overview of C#**

- Structure of a C# Program
- Basic Input/Output Operations
- Recommended Practices
- Compiling, Running, and Debugging

## **Module 3: Using Value-Type Variables**

- Common Type System
- Naming Variables
- Using Built-In Data Types
- Creating User-Defined Data Types
- Converting Data Types

## **Module 4: Statements and Exceptions**

- Introduction to Statements
- Using Selection Statements
- Using Iteration Statements
- Using Jump Statements
- Handling Basic Exceptions
- Raising Exceptions

## **Module 5: Methods and Parameters**

- Using Methods
- Using Parameters
- Using Overloaded Methods

## **Module 6: Arrays**

- Overview of Arrays
- Creating Arrays
- Using Arrays

## **Module 7: Essentials of Object-Oriented Programming**

- Classes and Objects

- Using Encapsulation
- C# and Object Orientation
- Defining Object-Oriented Systems

#### **Module 8: Using Reference-Type Variables**

- Using Reference-Type Variables
- Using Common Reference Types
- The Object Hierarchy
- Namespaces in the .NET Framework
- Data Conversions

#### **Module 9: Creating and Destroying Objects**

- Using Constructors
- Initializing Data
- Objects and Memory
- Resource Managements

#### **Module 10: Inheritance in C#**

- Deriving Classes
- Implementing Methods
- Using Sealed Classes
- Using Interfaces
- Using Abstract Classes

#### **Module 11: Aggregation, Namespaces, and Advanced Scope**

- Using Internal Classes, Methods, and Data
- Using Aggregation
- Using Namespaces
- Using Modules and Assemblies

#### **Module 12: Operators and Events**

- Introduction to Operators
- Operator Overloading

- Creating and Using Delegates
- Defining and Using Events

**Module 13: Properties and Indexers**

- Using Properties
- Using Indexers

**Module 14: Attributes**

- Overview of Attributes
- Defining Custom Attributes
- Retrieving Attribute Values

**Upcoming Classes**

Jun 04, 2012 - Jun 08, 2012  
Jul 16, 2012 - Jul 20, 2012  
Aug 27, 2012 - Aug 31, 2012  
Sep 17, 2012 - Sep 21, 2012  
Oct 15, 2012 - Oct 19, 2012  
Nov 19, 2012 - Nov 23, 2012  
Dec 10, 2012 - Dec 14, 2012