

CS214: C# Programming with Visual Studio

The course begins by providing a look at the different components of the .NET framework and discusses the basic building blocks of the C# language including variables, conditionals and looping. Next, Object-oriented features found in C# are discussed so that students understand how to leverage concepts such as inheritance, method overriding, interfaces, and object hierarchies. The course also shows several of the exciting language features available in C# including lambda expressions and LINQ to Objects. Other topics covered include data access technologies, lambda expressions as well as introductory look at building ASP.NET applications.



AUDIENCE

This course is valuable for developers who are interested in building applications using the C# programming language.

PREREQUISITES

This course is designed for software developers with previous programming experience using a programming language such as Visual Basic, C++, or Java.

WHAT YOU WILL LEARN

After attending this course, students will know:

- » C# Language Fundamentals
- » Object-oriented principles
- » C# Generics and other C# features
- » Visual Studio features
- » The role interfaces play in .NET
- » The role of classes, objects, namespaces, and assemblies
- » .NET Base Class Libraries
- » Components of the .NET framework
- » Data access with ADO.NET
- » Working with Multiple Threads
- » LINQ and Lambdas
- » Fundamentals of using C# in ASP.NET Application

“Interface is amazing! I’ll definitely come back for any training I need.”

Interface Student
Phoenix, AZ

\$2295.00

- 5-day course
- Compatible with MOC 20483
- Promo and package discounts may apply

QUESTIONS?
Call 602-266-8585



CAN'T MAKE IT TO CLASS IN PERSON?

Attend many classes online with RemoteLive.™

Call 602-266-8585 today for a live demo.

©2015 Interface Technical Training All rights reserved

(course outline
on back side)



COURSE OUTLINE

C# Programming with Visual Studio

1. Getting Started with the .NET Framework

- » What is the .NET Framework?
- » The role of the Common Language Runtime (CLR)
 - What is the CLR and what does it do in .NET?
 - .NET object-oriented language choices
 - Multi-language interoperability
 - Memory Management and Garbage Collection
- » .NET Compilation Model
- » Microsoft Intermediate Language (MSIL)
- » Just-in-Time compilation
- » Assemblies and Namespaces
- » The Base Class Library

2. C# Fundamentals

- » Compiling C# applications
- » Key Features in C#
 - Type-safety
 - Structured exception handling
 - Inheritance
 - Method Overloading/Overriding
 - Events
 - Multi-Threading

3. Visual Studio

- » Visual Studio Windows and Panels
- » Understanding Solutions and Projects
- » Visual Studio Feature Overview
- » Using Debugger Windows

4. C# Language Syntax

- » C# Fundamentals
 - 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
- » Type Casting
- » if..else statements
- » ? and : operators
- » switch statements
- » looping statements
- » Conditionals and Looping

5. Classes and Structs

- » What are classes?
- » What are objects?
- » Class members
- » Creating a class from scratch
- » Setting class member access visibility
- » Adding constructors
- » Adding fields to a class
- » Adding properties
- » Adding methods
- » The Role of Namespaces
- » What is a struct?
- » Differences between classes and structs
- » Creating a class in VS with the Class Designer

6. Object-Oriented Programming

- » What is object-oriented programming?
- » The role of System.Object in .NET
- » Understanding Abstraction, Encapsulation, Polymorphism and Inheritance
- » Using abstract classes
- » Overriding members
- » Understanding casting, boxing and unboxing in .NET
- » Structured Exception handling
- » Creating a finally block

7. Generics

- » What are Generics?
- » Creating objects using Generics
- » Defining a custom Generic type
- » Generic Constraints
- » Using the Nullable Struct
- » The default keyword
- » System.Collections.Generic Classes

8. Working with Interfaces

- » What are Interfaces?
- » Defining interfaces
- » Implementing interfaces
- » Interfaces and Polymorphism
- » Interfaces in the .NET Framework
- » Implementing IEnumerable

9. Using the Base Class Library

- » Overview of functionality in the framework class library
- » Working with System.IO classes
- » Working with Dates and Times
- » Accessing remote data and sending email with System.Net classes
- » Building strings with the StringBuilder class
- » Pattern searching with Regular Expressions
- » Working with Threads

10. Using Language Integrated Query (LINQ)

- » What is Language Integrated Query (LINQ)
- » The role of anonymous types
- » LINQ Expressions
- » Lambda Expressions
- » Using LINQ to Objects

11. Data Access Technologies

- » Introduction to ADO.NET
- » What's new in ADO.NET?
- » Managed Provider Classes in ADO.NET
- » Generic database access with DbProviderFactory classes
- » Multiple Active Resultsets (MARS)
- » Introduction to Object-Relational-Modeling (ORM) Frameworks
 - LINQ to SQL
 - Entity Framework

12. Introduction to ASP.NET Programming with C#

- » ASP.NET Web Form Features
- » The Role of the Page Class
- » Handling Events
- » Using Server Controls
- » Creating and Using Master Pages