

Object-Oriented Programming Using C# 2012 Level 1

Course ISI-1399 Five Days - Instructor-led - Hands on

Introduction

This course is appropriate to those who are new to Microsoft's .NET development environment or if you're an experienced .NET developer who's new to C#. It is also appropriate if you have programming experience with another language like Java, C++, or COBOL.

This course starts you off using all the best features of Visual Studio 2012 so you can begin productive work almost immediately. Object-oriented programming is made clear as business objects are presented as customers, invoices and products, so you can see how OOP is used to create multi-tiered applications in the real world

The course builds your skills in incremental steps to teach you how to validate data, handle numeric, date and string data, work with loops, arrays and collections, code methods and event handlers, handle exceptions, work with text, binary and XML data.

After completing the two courses in this series, students will find that object oriented programming is made clear. Business objects are described so that you can see how OOP is used to create multi-tiered applications in the real world.

At Course Completion

After completing this course, students will be able to understand all the new language features that make C# such as outstanding development tool:

Get going fast with the C# essentials:

- Create your first Windows Forms applications using C# 2012, Visual Studio 2012 and the .NET Framework
- Master essential C# programming skills, including
 - Work with numeric and string data
 - Code control structures, methods and event handlers
 - Handle exceptions and validate data
 - Use arrays and collections
- Work with dates and strings
- Gain the debugging skills to track down elusive coding errors

Prerequisites

Before attending this course, students should have the following skills and knowledge:

- This course is appropriate for students new to the .NET environment
- This course is appropriate for students who have programming experience with another language like Java, C++ or COBOL.

Student Materials

The student kit includes a workbook and other necessary materials for this class.

Course Outline

Introduction to Visual Studio

Module 1: How to get started with Visual Studio

This module introduces you to the basics of working with C# 2010, the .NET Framework, and Visual Studio. This is the background that you need for getting started right with C#

- An introduction to .NET development
- A tour of the Visual Studio IDE
- How to test a project
- How to convert projects and change .NET Framework versions

Module 2: How to design a Windows Forms application

Learn how to use Visual Studio to design a Windows form that contains labels, text boxes, and buttons. This form will get user input, do two calculations on it, and display the results, so it's not just a "Hello World!" application. When you finish this module, you'll be ready to learn how to write the C# code for the form.

- How to set options and create a new project
- How to design a form
- How to name and save the files of a project

Module 3: How to code and test a Windows Forms application

This module shows you how to write the C# code for the Windows form that was designed in module 2. This module also gets you started with practical debugging techniques. When you finish this module, you'll be able to use Visual Studio to design a form; write the C# code for the form; and debug the form

- An introduction to coding
- How to add code to a form
- More coding skills
- How to run, test and debug a project

The C# Language Essentials

Module 4: How to work with numeric and string data

This module shows you how to perform arithmetic operations on the various types of numeric data and how to work with string data. Beyond that, this module presents three important: casting, data conversion, and scope.

- How to work with numeric and string data

- How to work with built-in value types
- How to work with strings
- How to convert data types
- Three other skills for working with data
- Two versions of the Invoice Total application

Module 5: How to code control structures

First, this module shows you how to control program execution and repetitive processing by using conditional statements and various types of loops. Then, it shows you some special debugging techniques for programs with loops in the context of a new application.

- How to code control structures
- How to code Boolean expressions
- How to code conditional statements
- How to code loops
- The Future Value application

Module 6: How to code methods and event handlers

This module shows you how to code and call procedures and how to generate event handlers for all types of events. Then it shows you how delegates are used to wire an event to a method so you can change that wiring whenever you need to.

- How to code methods and event handlers
- How to code and call methods
- How to work with events and delegates
- Another version of the Future Value application

Module 7: How to handle exceptions and validate data

First, this module shows you how to use structured exception handling so you can create bulletproof applications. Then, it shows you the best ways to validate the data that the user enters so exceptions are avoided.

- An introduction to exceptions
- How to use structured exception handling
- How to validate data
- The Future Value application with data validation

Module 8: How to work with arrays and collections

First, this module shows you how to use three types of arrays. Then, it shows you how to use the methods and properties of the Array class for working with an array. Last, it shows you how to use collection classes to work with typed and untyped collections

- How to work with one-dimensional arrays
- How to work with rectangular arrays
- How to work with jagged arrays
- More skills for working with arrays
- How to work with collections

Module 9: How to work with dates and strings

First, this module shows you how to use the properties and methods of the `DateTime` structure to work with dates and times. Then, this module shows you how to use the properties and methods of the `String` and `StringBuilder` classes to manipulate the data in strings, to validate user entries, and to format numbers, dates, and times.

- How to work with dates and times
- How to work with strings
- How to format numbers, dates and times

Module 10: More skills for working with controls and multi-form projects

To keep the emphasis on the C# essentials, the first nine modules present single-form applications that contain label, text box, and button controls. But now that you're ready for it, this module shows you how to use controls like combo boxes, list boxes, and check boxes in developing multi-form applications.

- How to work with controls
- How to work with multi-form projects
- How to use the `MessageBox` class
- The Payment application

Module 11: How to debug an application

Modules 3 and 5 present some simple debugging techniques. But now that you know more about C#, this module presents the most useful debugging features that Visual Studio has to offer. When you finish this module, you'll have all the skills you need for debugging complex applications.

- Basic debugging techniques
- How to use the debugging windows

Object-Oriented Programming

Module 12: How to create and use classes

This module shows you how to create your own C# classes that include fields, properties, methods, constructors, and static methods. This course uses real-world business objects such as products and customers. In fact, this module presents an application that uses a `Product` class to create `Product` objects and a `ProductDB` class that uses static methods to handle the application's database access. That's the type of complete application that you need to see if you want to understand object-oriented programming in C#.

- An introduction to classes
- How to create a class
- The Product Maintenance application
- How to use the Class View window and class diagrams

How to work with struc