

Programming in C# with Microsoft Visual Studio 2010



Overview

The course focuses on C# program structure, language syntax, and implementation details with .NET Framework 4.0. This course describes the new enhancements in the C# 4.0 language by using Visual Studio 2010.

In this course, lower-intermediate level programmers gain the knowledge and skills they need to develop C# applications for the Microsoft .NET Framework 4.0. The course highlights the structure of C# 4.0 programs, language syntax, and implementation details. This course is not mapped to any exam.

Prerequisites

Before attending this course, students must have:

- At least 12 months experience working with an Object Oriented language
- Have C++ or Java knowledge:
 - Creating Classes
 - Inheritance and Abstraction
 - Polymorphism
 - Interfaces
 - Exceptions
- Knowledge of the Visual Studio IDE.

Course outline

Module 1: Introducing C# and the .NET Framework

- Introduction to the .NET Framework
- Creating Projects Within Visual Studio 2010
- Writing a C# Application
- Building a Graphical Application
- Documenting an Application
- Running and Debugging Applications by Using Visual Studio 2010

Module 2: Using C# Programming Constructs

- Declaring Variables and Assigning Values
- Using Expressions and Operators
- Creating and Using Arrays
- Using Decision Statements
- Using Iteration Statements

Module 3: Declaring and Calling Methods

- Defining and Invoking Methods
- Specifying Optional Parameters and Output Parameters

Module 4: Handling Exceptions

- Handling Exceptions
- Raising Exceptions

Module 5: Reading and Writing Files

- Accessing the File System
- Reading and Writing Files by Using Streams

Module 6: Creating New Types

- Creating and Using Enumerations
- Creating and Using Classes
- Creating and Using Structs
- Comparing References to Values

Module 7: Encapsulating Data and Methods

- Controlling Visibility of Type Members
- Sharing Methods and Data

Module 8: Inheriting From Classes and Implementing Interfaces

- Using Inheritance to Define New Reference Types
- Defining and Implementing Interfaces
- Defining Abstract Classes

Module 9: Managing the Lifetime of Objects and Controlling Resources

- Introduction to Garbage Collection
- Managing Resources

Module 10: Encapsulating Data and Defining Overloaded Operators

- Creating and Using Properties
- Creating and Using Indexers
- Overloading Operators

Module 11: Decoupling Methods and Handling Events

- Declaring and Using Delegates
- Using Lambda Expressions
- Handling Events

Module 12: Using Collections and Building Generic Types

- Using Collections
- Creating and Using Generic Types
- Defining Generic Interfaces and Understanding Variance
- Using Generic Methods and Delegates

Module 13: Building and Enumerating Custom Collection Classes

- Implementing a Custom Collection Class
- Adding an Enumerator to a Custom Collection Class

Module 14: Using LINQ to Query Data

- Using the LINQ Extension Methods and Query Operators
- Building Dynamic LINQ Queries and Expressions

Module 15: Integrating Visual C# Code with Dynamic Languages and COM Components

- Integrating C# Code with Ruby and Python
- Accessing COM Components from C#