M55339A - Programming with C#

DURATION

5 Days

LEVEL

200 - Intermediate

VENDOR / PRODUCT

Development

LEGACY MOC CODE

M20483C

COURSE OVERVIEW

This training course teaches developers the programming skills that are required for developers to create applications using the C# language. During their five days in the classroom students review the basics of C# program structure, language syntax, and implementation details, and then consolidate their knowledge throughout the week as they build an application that incorporates several features of the .NET 6.0. The course aims to follow the spirit of the Microsoft Official Curriculum course 20483, while bringing it completely up-to-date with the latest features of Visual Studio 2022, and the cross-platform capabilities of .NET 6.0.

Labs available from go deploy.

AUDIENCE PROFILE

This course is intended for experienced developers who already have programming experience in C, C++, JavaScript, Objective-C, Microsoft Visual Basic, or Java and understand the concepts of object-oriented programming. This course is not designed for students who are new to programming; it is targeted at professional developers with at least one month of experience programming in an object-oriented environment. Those new to programming should consider course *55337AC – Introduction to Programming*. The *55337AC* course uses C# as the language to facilitate an introduction to programming generally, whereas this course focuses on the C# language itself, making it an excellent follow-on course. If you want to learn to take full advantage of the C# language, then this is the course for you.

This is student material. Instructors, please contact the publisher at https://cwlink.sc/InstructorAccess.

COURSE OUTLINE

Module 1: C# Syntax

Microsoft .NET 6 provides a comprehensive development platform that you can use to build, deploy, and manage applications and services. By using .NET, you can create visually compelling applications, enable seamless communication across technology boundaries, and provide support for a wide range of business processes.

In this module, you'll learn about some of the core features provided by.NET and Microsoft Visual Studio. You'll also learn about some of the core C# constructs that enable you to start developing .NET applications.

Lessons

- Lesson 1: Writing Applications in C# and .NET
- Lesson 2: Types of Data and Expressions
- Lesson 3: C# Language Constructs

Module 2: C# Language Concepts

Applications often consist of logical units of functionality that perform specific functions, such as providing access to data or triggering some logical processing. C# is an object-orientated language and uses the concept of methods to encapsulate logical units of functionality. Although good practice is to have methods that do just one thing, they can be as simple or as complex as you like. It is also important to consider what happens to the state of your application when an exception occurs in a method.

In this module, you'll learn how to create and use methods and how to handle exceptions. You'll also learn how to use logging and tracing to record the details of any exceptions that occur.

Lessons

- Lesson 1: Methods
- Lesson 2: Method Overloading
- Lesson 3: Exception Handling
- Lesson 4: Monitoring

Module 3: C# Structures, Collections and Events

To create effective applications, you must first learn some fundamental C# constructs. You need to know how to create simple structures to represent the data items you are working with. You need to know how to organize these structures into collections, so that you can add items, retrieve items, and iterate over your items. Finally, you need to know how to subscribe to events so that you can respond to the actions of your users.

In this module, you'll learn how to create and use structs and enums, organize data into collections, and create and subscribe to events.

Lessons

- Lesson 1: Structs
- Lesson 2: Enums
- Lesson 3: Built-in Collections

- Lesson 4: Events

Module 4: C# Classes

In this module, you'll learn how to use interfaces and classes to define and create your own custom, reusable types. You'll also learn how to create and use enumerable type-safe collections of any type.

Lessons

- Lesson 1: Creating Classes
- Lesson 2: Interfaces
- Lesson 3: Understanding Generics in C#

Module 5: C# Inheritance

In this module, you'll learn how to use inheritance to create class hierarchies and to extend .NET types.

Lessons

- Lesson 1: Hierarchies of Classes
- Lesson 2: Polymorphism
- Lesson 3: Extending Classes

Module 6: Input and Output

In this module, you'll learn how to read and write data by using transactional filesystem I/O operations, how to serialize and deserialize data to the filesystem, and how to read and write data to the filesystem by using streams.

Lessons

- Lesson 1: File I/O
- Lesson 2: Serialization and Deserialization
- Lesson 3: Streams

Module 7: Database Access

In this module, you'll learn how to use Entity Framework and how to query many types of data by using Language-Integrated Query (LINQ).

Lessons

- Lesson 1: Entity Framework
- Lesson 2: LINQ

Module 8: Using the Network

In this module, you'll learn how to use the request and response classes in the System.Net namespace to directly manipulate remote data sources. You'll also learn about REST and OData and look briefly at ASP.NET Core MVC.

Lessons

- Lesson 1: Web Services
- Lesson 2: REST and OData
- Lesson 3: ASP.NET Core MVC

Module 9: Graphical User Interfaces

In this module, you'll learn how to use Extensible Application Markup Language (XAML) and Windows Presentation Foundation (WPF) to create engaging UIs.

Lessons

- Lesson 1: Using UI Frameworks
- Lesson 2: Data binding
- Lesson 3: Styling the UI

Module 10: Application Performance

In this module, you'll learn how to improve the performance of your applications by distributing your operations across multiple threads.

Lessons

- Lesson 1: Multitasking
- Lesson 2: Asynchronous Calls
- Lesson 3: Dealing with Conflicts

Module 11: C# Interop

In this module, you'll learn how to interoperate with unmanaged code in your applications, and how to ensure that your code releases any unmanaged resources.

Lessons

- Lesson 1: Dynamic Objects
- Lesson 2: Managing Resources

Module 12: Designing for Reuse

In this module, you'll learn how to consume existing assemblies by using reflection, and how to add additional metadata to types and type members by using attributes. You'll also learn how to generate code at runtime by using the Code Document Object Model (CodeDOM) and how to manage your .NET assemblies.

Lessons

- Lesson 1: Metadata
- Lesson 2: Attributes

- Lesson 3: Generating Code
- Lesson 4: Assemblies

Module 13: Securing Data

In this module, you’ll learn how to use the cryptography classes provided by .NET. You’ll implement symmetric and asymmetric encryption, and learn how to use hashes to generate mathematical representations of your data. You’ll also learn how to create and manage X509 certificates and how to use them in the asymmetric encryption process.

Lessons

- Lesson 1: Application Security
- Lesson 2: Symmetric Encryption
- Lesson 3: Asymmetric Encryption