

Programming in C#

20483B



Delivery Type: Classroom

Duration: 5 days

Overview:

This training course teaches developers the programming skills that are required for developers to create Windows 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 Framework 4.5. The course introduces many of the techniques and technologies employed by modern desktop and enterprise applications. At the end of the course, students should leave the class with a solid knowledge of C# and how to

use it to develop .NET Framework 4.5 applications.

This course uses Visual Studio 2012, running on Windows 8.

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.

Course Completion:

After completing this course, students will be able to:

- ✓ Describe the core syntax and features of C#.
- ✓ Create and call methods, catch and handle exceptions, and describe

- the monitoring requirements of large-scale applications.
- ✓ Implement the basic structure and essential elements of a typical desktop application.
 - ✓ Create classes, define and implement interfaces, and create and use generic collections.
 - ✓ Use inheritance to create a class hierarchy, extend a .NET Framework class, and create generic classes and methods.
 - ✓ Read and write data by using file input/output and streams, and serialize and deserialize data in different formats.
 - ✓ Create and use an entity data model for accessing a database and use LINQ to query and update data.
 - ✓ Use the types in the System.Net namespace and WCF Data Services to access and query remote data.
 - ✓ Build a graphical user interface by using XAML.
- ✓ Improve the throughput and response time of applications by using tasks and asynchronous operations.
 - ✓ Integrate unmanaged libraries and dynamic components into a C# application.
 - ✓ Examine the metadata of types by using reflection, create and use custom attributes, generate code at runtime, and manage assembly versions.
 - ✓ Encrypt and decrypt data by using symmetric and asymmetric encryption.

Prerequisites:

Developers attending this course should already have gained some limited experience using C# to complete basic programming tasks. More specifically, students should have hands-on experience using C#.