Programming in C#

Course ID#: 1411-060-C#-W

35 Hrs

Course Content

Course Description:

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.

Prerequisites:

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.

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# that demonstrates their understanding of the following:

- How to name, declare, initialize and assign values to variables within an application.
- How to use:
- arithmetic operators to perform arithmetic calculations involving one or more variables;
- relational operators to test the relationship between two variables or expressions;
- logical operators to combine expressions that contain relational operators.
- How to create the code syntax for simple programming statements using C# language keywords and recognize syntax errors using the Visual Studio IDE.

·How to create a simple branching structure using an IF statement.

- ·How to create a simple looping structure using a For statement to iterate through a data array.
- ·How to use the Visual Studio IDE to locate simple logic errors.
- ·How to create a Function that accepts arguments (parameters and returns a value of a specified type.
- ·How to design and build a simple user interface using standard controls from the Visual Studio toolbox.
- ·How to connect to a SQL Server database and the basics of how to retrieve and store data.
- ·How to sort data in a loop.
- ·How to recognize the classes and methods used in a program.

Topics

Module 1: Review of C# Syntax

Lessons

- Overview of Writing Applications using C#
- Datatypes, Operators, and Expressions
- C# Programming Language Constructs

Lab : Developing the Class Enrolment Application

- Implementing Edit Functionality for the Students List
- Implementing Insert Functionality for the Students List
- Implementing Delete Functionality for the Students List
- Displaying the Student Age

.

Module 2: Creating Methods, Handling Exceptions, and Monitoring Applications

Lessons

- Creating and Invoking Methods
- Creating Overloaded Methods and Using Optional and Output Parameters
- Handling Exceptions
- Monitoring Applications

Lab : Extending the Class Enrolment Application Functionality

- Refactoring the Enrolment Code
- Validating Student Information
- Saving Changes to the Class List

Module 3: Developing the Code for a Graphical Application Lessons

- Implementing Structs and Enums
- Organizing Data into Collections
- Handling Events

Lab : Writing the Code for the Grades Prototype Application

- Adding Navigation Logic to the Application
- Creating Data Types to Store User and Grade Information
- Displaying User and Grade Information

Module 4: Creating Classes and Implementing Type-safe Collections

Lessons

- Creating Classes
- Defining and Implementing Interfaces
- Implementing Type-safe Collections

Lab : Adding Data Validation and Type-safety to the Grades Application

- Implementing the Teacher, Student, and Grade Types as Classes
- Adding Data Validation to the Grade Class
- Displaying Students in Name Order
- Enabling Teachers to Modify Class and Grade Data

Module 5: Creating a Class Hierarchy by Using Inheritance

Lessons

- Creating Class Hierarchies
- Extending .NET Framework Classes
- Creating Generic Types

Lab : Refactoring Common Functionality into the User Class

- Creating and Inheriting from the User Base Class
- Implementing Password Complexity by Using an Abstract Method

- Creating the ClassFullException Class

Module 6: Reading and Writing Local Data

Lessons

- Reading and Writing Files
- Serializing and Deserializing Data
- Performing I/O Using Streams

Lab : Generating the Grades Report

- Serializing the Data for the Grades Report as XML
- Previewing the Grades Report
- Persisting the Serialized Grades Data to a File

Module 7: Accessing a Database

Lessons

- Creating and Using Entity Data Models
- Querying Data by Using LINQ
- Updating Data by Using LINQ

Lab : Retrieving and Modifying Grade Data

- Creating an Entity Model from the The School of Fine Arts Database
- Updating Student and Grade Data Using the Entity Framework
- Extending the Entity Model to Validate Data

Module 8: Accessing Remote Data

Lessons

- Accessing Data Across the Web
- Accessing Data in the Cloud

Lab : Retrieving and Modifying Grade Data in the Cloud

- Creating a WCF Data Service for the SchoolGrades Database
- Integrating the WCF Data Service into the Application
- Retrieving Student Photographs Over the Web (if time permits)

Module 9: Designing the User Interface for a Graphical Application

Lessons

- Using XAML to Design a User Interface
- Binding Controls to Data
- Styling a User Interface

Lab : Customizing Student Photographs and Styling the Application

- Customizing the Appearance of Student Photographs
- Styling the Logon View
- Animating the StudentPhoto Control (If Time Permits)

Module 10: Improving Application Performance and Responsiveness

Lessons

- Implementing Multitasking by using Tasks and Lambda Expressions
- Performing Operations Asynchronously
- Synchronizing Concurrent Access to Data

Lab : Improving the Responsiveness and Performance of the Application

- Ensuring that the User Interface Remains Responsive When Retrieving Data for Teachers
- Providing Visual Feedback During Long-Running Operations

Module 11: Integrating with Unmanaged Code

Lessons

- Creating and Using Dynamic Objects
- Managing the Lifetime of Objects and Controlling Unmanaged Resources

Module 11: Continued

Lab : Upgrading the Grades Report

- Generating the Grades Report by Using Microsoft Office Word
- Controlling the Lifetime of Word Objects by Implementing the Dispose Pattern

Module 12: Creating Reusable Types and Assemblies

Lessons

- Examining Object Metadata
- Creating and Using Custom Attributes
- Generating Managed Code
- Versioning, Signing and Deploying Assemblies

Lab : Specifying the Data to Include in the Grades Report

- Creating the IncludeInReport Attribute
- Generating the Report
- Storing the Grades.Utilities Assembly Centrally

Module 13: Encrypting and Decrypting Data

This module explains how to encrypt and decrypt data by using symmetric and asymmetric encryption.

Lessons

- Implementing Symmetric Encryption
- Implementing Asymmetric Encryption

Lab : Encrypting and Decrypting Grades Reports

- Encrypting the Grades Report
- Decrypting the Grades Report