Windows Forms Programming Using C#

Duration: 35 hours

Prerequisites: Prior programming experience is required.

Course Description: This course provides students with hands on experience using Visual Studio to create desktop applications using Windows Forms and the .NET Framework using C#. The course provides a thorough introduction to the C# programming language, including coverage of the essentials of the C# programming language, built in data types, operators, control structures, classes and methods, collections and exception handling.

Students then learn how to leverage the power of the .NET Framework to build desktop applications. Students learn how to build Windows Forms applications and use with a variety of controls to create sophisticated user interfaces. Students also learn how to use the BackgroundWorker to perform asynchronous operations.

Students also learn how to use ADO.NET to interact with databases and XML files. Students learn how Windows Forms uses data binding to display data in controls such as the DataGridView and Chart. Students also learn how to build and interact with simple WCF SOAP Web Services.

Other topics include: debugging techniques; using a .config file to control application configuration; building menus, toolbars and status bars; reading and writing files; interacting with the file system; and deploying desktop applications.

Comprehensive labs provide the students with extensive experience creating and deploying Windows Forms-based desktop applications.

Students Will Learn

- Introduction to .NET
- Using Visual Studio
- C# Basics
- Introduction to Object-Oriented Programming
- Control Structures, Methods and Exceptions
- Working with Data Collections
- User-Defined Data Types
- Introduction to Windows Forms
- Controls and Event Handling
- Additional Controls
- ADO.NET: Connected Access
- ADO.NET: Disconnected Access
- Data Binding
- Modal and Modeless Forms
- Using the BackgroundWorker
- Using Menus, Toolbars and Status Bars
- WCF Services
- File I/O
- Deploying Projects

Overview

Introduction to .NET

- Overview of the .NET Framework
- How .NET is Different from Traditional Programming
- Common Language Runtime (CLR)
- Common Language Specification (CLS)
- Common Type System (CTS)
- .NET Assemblies
- Microsoft Intermediate Language (CIL)
- .NET Namespaces
- .NET Framework Class Library

Introduction to Visual Studio

- Creating a Project
- Using the Code Editor
- Correcting Syntax Errors
- Setting Project Properties
- Adding References
- Compiling a Program
- Running a Program
- Debugging a Program
- Using the MSDN (Help)

Language Fundamentals

- C# Program Structure
- Defining Namespaces
- Understanding C# Data Types
- Defining Variables and Constants
- Comparing Value Types vs. Reference Types
- Working with Operators and Expressions
- Performing Type Conversions
- Using Console I/O
- Formatting Numbers, Date and Times

Conditionals and Looping

- `if/else`
- `switch`
- `while` and `do/while`
- `for`
- `foreach`

Methods and Parameters

- Defining Static and Instance Methods
- Passing Parameters by value and by reference
- Overloading Methods
- Using Variable Length Parameter Lists

Exception Handling

- What are Exceptions?
- .NET Exception Hierarchy
- Catching Exceptions
- Throwing Exceptions
- Managing Resources with Finally

Collections

- Defining and Using Arrays
- Understanding `System.Array`
- .NET Collections vs Generic Collections
- Working with Lists
- Working with Dictionaries
- Using LINQ to Objects

Object-Oriented Programming

- Overview of Object-Oriented Programming
- Building Classes
- Defining Properties
- Using Auto-Implemented Properties
- Defining Methods
- Understanding Constructors
- Extending .NET Classes via Inheritance
- Defining and Implementing Interfaces
- Understanding the Role of Interfaces in .NET

Windows Forms Applications

- Windows Forms Applications
- Setting Form Properties
- Understanding the Life-cycle of a Form
- Using the Windows Forms Designer
- Using the `MessageBox` Class
- Using a `.config` File

Using Controls

- Working with Windows Forms Controls
- Using `Text` Controls
- Using `Button` Controls
- Using `Selection` Controls
- Using `List` Controls
- Using `Container` Controls
- Using `Image` Controls
- Using `Up/Down` Controls

- Using the `ErrorProvider` and `ToolTipProvider` Controls

Handling Events

- Understanding the Event-Driven Programming Model
- Writing Event Handlers
- Sharing Event Handlers

Performing Asynchronous Activities

- Understanding Threading
- Working with Delegates
- Using the Background Worker
- Updating Controls from Other Threads

ADO.NET

- Understanding the ADO.NET Object Model
- Connected vs. Disconnected Access
- Using a Connection to Connect to a Data Source
- Using a Command to Execute Queries and Stored Procedures
- Using a `DataReader` to Work with Cursors
- Using the `DataSet` with Disconnected Data
- Using `DataAdapters` with DataSets

Using XML

- Understanding XML and XML Schemas
- Reading XML Data with a `DataSet`
- Writing XML Data with a `DataSet`

Data Binding

- Understanding ADO.NET Data Binding
- Binding to Simple and Complex Controls
- Manually Binding Controls
- Using the `BindingSource` Control
- Using the `BindingNavigator` Control
- Using the `DataGridView` Control
- Using the `Chart` Control

Working with Forms

- Understanding Modal vs Modeless Forms
- Displaying Modal Forms
- Working with `DialogResult`
- Retrieving Data from Modal Forms
- Displaying Modeless Forms
- Working with Data in Modeless Forms
- Using the Common Dialogs

Working with Menus, Toolbars and Status Bars

- Working with Menus
- Working with Toolbars
- Working with Status Bars

Working with Files and Directories

- Working with the Windows File System
- Discovering Drives
- Discovering Directories
- Discovering Files
- Examining File, Directory and Drive Attributes
- Reading, Writing and Appending to Files

Introduction to WCF Web Services

- Overview of WCF Services
- Understanding SOAP
- Creating a WCF SOAP Web Service
- Creating a Proxy Class
- Calling a WCF SOAP Web Service

Deploying Windows Forms Applications

- Understanding Deployment Options
- Configuring an Application for Deployment
- Using XCOPY Deployment
- Using Installers
- Using Click-Once Deployment

Related Bootcamp

Track	Duration	Price
Microsoft .NET Developer: C#	161 hours	$4,475

Contact Us

Address: 6 Hemlock Drive Chelmsford, MA 01824

Phone: 978.250.4983

Mon - Thur: 9 am - 5 pm EST
Fri: 9 am - 4 pm EST

E-mail: info@developer-bootcamp.com

Copyright© 2025 Developer Bootcamp