

## Waters – C# .Net Training

5 Days

### **COURSE OVERVIEW**

It is anticipated that to cover the content requested the course will be an intensive hand on experience.

### **COURSE CONTENT**

#### **.NET and C# Building Blocks**

- Overview of the .NET platform, and VS.NET IDE, CLR and FCL
- Assemblies, metadata, namespaces, assembly info, managed execution
- Using the System Console for I/O, formatting values
- Basic data types, reference vs value types, variable scoping
- Operators; numeric parsing, DateTime

#### **Basic Constructs**

- Looping and branching: if, switch, while, do, for, foreach
- Exception handling
- Arrays, single- and multi-dimensional, jagged arrays; the System Array type
- String and StringBuilder, regular expressions
- Functions/Methods and arguments

#### **Object-Oriented Programming**

- Classes; data members (fields), private and public access
- Constructors, this, overloading constructors, initializer lists; constant and readonly fields
- Class member functions, static functions, ref and out parameters, variable-number params
- Static data and function members, static constructors; function overloading
- The object type; GetType and ToString
- Properties: getting and setting values; readonly and static properties
- Structs vs Classes

#### **Inheritance**

- Inheritance; protected members; hiding inherited functions with new
- Virtual functions and overrides, calling base functions
- Abstract classes and properties, sealed classes; nested classes
- Properties and inheritance; virtual properties; object overrides

#### **Class Relationships**

- Internal members; internals and assemblies; protected internals, aggregation/association
- Namespaces and assemblies, using aliases, common language
- System collection classes: ArrayList, Queue, Stack, Hashtable

## Interfaces

- Interfaces; interfaces vs abstract classes
- is and as operators; implementation and interface inheritance
- Multiple interfaces; hiding interfaces; interface properties
- IEnumerator, IComparable, IComparer

## Operators and Conversions

- Numeric conversions; checked and unchecked, implicit and explicit conversions
- Conversion of structs and classes; object params
- User-defined conversions; operator overloading: equivalence; enums

## Indexers, Delegates and Events

- Indexing with integers; integer and string indexers; indexers and foreach; enumerators
- Creating and using delegates
- Defining and using events; EventArgs
- Asynchronous calls

## I/O

- Streams I/O, file, memory and buffered streams, stream readers/writers
- File and directory manipulation
- Serialization; implementing ISerializable

## Creating an Application by Using WPF

This module explains how to build a WPF application.

- Overview of WPF
- Creating a Simple WPF Application
- Handling Events and Commands
- Navigating Between Pages

## Building User Interfaces

This module explains how to build a user interface in a WPF application.

- Defining Page Layout
- Building User Interfaces by Using Content Controls
- Building User Interfaces by Using Items Controls
- Hosting Windows Forms Controls

## Customizing Appearance

This module explains how to customize the appearance of a WPF application.

- Sharing Logical Resources in an Application
- Creating Consistent User Interfaces by Using Styles
- Changing the Appearance of Controls by Using Control Templates
- Enhancing User Interfaces by Using Triggers and Animations

## Data Binding

This module explains how to bind user interface controls to data sources.

- Overview of Data Binding
- Creating a Data Binding
- Implementing Property Change Notification
- Converting Data
- Validating Data

## Data Binding to Collections

This module explains how to bind user interface controls to collections.

- Binding to Collections of Objects

- Presenting Data by Using Collection Views
- Presenting Data by Using Data Templates

### **Creating New Controls**

This module explains how to create new controls in a WPF application.

- Overview of Control Authoring
- Creating Controls

### **Managing Documents**

This module explains how to manage documents in a WPF application.

- Creating and Viewing Flow Documents
- Creating and Viewing Fixed Documents
- Packaging Documents
- Printing Documents

### **Getting Started with Windows Communication Foundation**

This module explains how to build a simple WCF service and client.

- Designing an Application to Be Part of a Service Oriented Architecture
- Overview of WCF Architecture
- Using a Language-Level Interface As a Service Contract
- Implementing a Simple WCF Service in Visual Studio 2008
- Consuming a simple WCF service in Visual Studio 2008

### **Configuring and Hosting WCF Services**

This module explains how to create and configure a WCF service as a managed application and select an appropriate hosting option.

- Programmatically Configuring a Managed Application to Host a WCF Service
- Programmatically Configuring a Managed Application to Call a WCF Service
- Defining Client and Service Settings by Using File-Based Configuration
- Selecting a Hosting Option for a WCF Service
- Deploying a WCF Service

### **Endpoints and Behaviors**

This module explains how to expose a WCF service over different endpoints and add run-time functionality by using behaviours.

- Exposing WCF Services Over Different Endpoints
- Adding Behaviors to Services and Endpoints
- Interoperating with Non-WCF Web services

### **Introduction to XML in the .NET Framework**

This module introduces the use of XML in the .NET Framework to help you solve common business problems. Overview

- Course at a Glance
- Common Business Problems
- XML in the .NET Framework

### **Parsing XML**

This module discusses how to parse XML data from a file, string, or stream by using the XmlReader class.

- Overview of XML Parsing
- Parsing XML Using XmlTextReader
- Creating a Custom Reader

### **Validating XML**

This module gives an overview of XML Schema Definition (XSD) data types, and then covers how to use the Microsoft .NET Framework XML classes to perform XML schema validation.

- Examining Schemas
- Validating XML While Parsing
- Advanced Validation
- Resolve a Uniform Resource Identifier (URI) reference to a schema or DTD.
- Create an application that uses the .NET Framework XML classes and an XSD schema or a DTD to validate well-formed XML.

### **Writing XML**

This module describes the .NET Framework classes that are used to write XML with or without namespaces.

- Overview of Generating XML Documents
- Writing XML
- Generating XML with Namespaces
- Controlling XML Format and Converting XML

### **Querying XML**

This module explains the process of querying an XML document by using XPath.

- Introduction to Querying XML Using XPath
- Creating and Navigating a Document Cache
- Executing Your Query

### **Manipulating Cached XML**

This module introduces the XML Document Object Model (XML DOM) and explains how to locate and manipulate XML by using the DOM.

- Introduction to the XML Document Object Model
- Navigating XML Using the DOM
- Creating New Nodes
- Modifying and Deleting Nodes

### **Transforming XML**

This module explains the process of transforming XML documents.

- Introduction to Transforming XML
- Programming the XsltTransform Class
- Extending XSLT Style Sheets

### **Serializing Objects as XML**

This module provides students with the skills needed to create XML documents from the data contained within programmatic objects. This process is referred to as serialization and can be used to persist object data or to share object data between applications.

- Introduction to XML Serialization
- Serializing Objects Using XmlSerializer
- Controlling XML Serialization

### **Maximising UI responsiveness with multiple threads**

- Spawning, detecting and aborting worker threads
- Simplifying multithreaded programs with the BackgroundWorker component
- Calling any function asynchronously with delegates

### **Component features of .NET**

- Manifests and assemblies

- Private vs. shared assemblies
- Deploying .NET components to the global assembly cache (GAC)
- Assembly versioning, use of strongnames etc

#### **Covered If Time allows:**

#### **Other stuff**

- Garbage collection : finalize v dispose etc.

#### **Printing and Reporting**

- Exploiting the .NET printing classes
- Manipulating the PrintDocument object
- Previewing output with printing controls

#### **Setting installation options**

- Launched vs. installed applications
- Application prerequisites
- Updating applications automatically
- Running programs via Internet Explorer

#### **Configuring Code Access Security (CAS)**

- Code groups and permission sets
- Running full or partially trusted code
- Setting ClickOnce security options

#### **Interop**

- Accessing COM/DCOM
- Tools for forwards and backwards compatibility
- Wrapping legacy components
- Platform Invoke
- c# Inter-Process-Comms

#### **The System.XML namespace**

- XmlReader and XmlWriter
- XmlTextReader and XmlTextWriter
- XmlValidatingReader
- XmlSerializer
- XmlDocument and XmlDataDocument
- XsltTransform