

Course 467: .NET Framework 2.0 Programming (5 days)

Course Description...

The .NET Framework provides an enormous amount of built-in functionality that developers can tap into to perform a variety of tasks including parsing flat-files or XML, securing applications, accessing databases, sending emails, dynamically inspecting assemblies and much more. The .NET Framework 2.0 Programming course provides developers with an in-depth look at many of the framework's key namespaces, classes, and interfaces and focuses on areas of the framework that can significantly enhance developer productivity. The course uses the C# programming language to demonstrate important .NET Framework concepts.

Topics covered include XML parsing APIs, Foundation Class Library classes, debugging and tracing, data serialization and deserialization, ADO.NET, XML Web Services plus more. An overview of .NET 3.0 technologies is also included to introduce students to new technologies that can be used to build rich desktop, workflow and distributed applications. Hands-on lab exercises allow students to enhance their skills by working directly with many aspects of the .NET Framework.

Learning Objectives...

- How to work with different Foundation Class Library classes
- How to inspect assemblies and types dynamically with reflection
- The role the StringReader and StringWriter classes play
- How to leverage Generics
- The role events, delegates and event handlers play in the framework
- How to create custom events and delegates
- How to create custom interfaces and implement framework interfaces
- How to serialize/deserialize data
- Options for parsing flat-files
- Classes and attributes that can simplify debugging
- How and why to add tracing functionality into applications
- The difference between symmetric and asymmetric encryption algorithms
- How to encrypt and decrypt data
- XML parsing APIs
- The role of .NET attributes in enhancing security
- The role of Code Access Security and how it works
- How to use ADO.NET classes to access databases
- .NET 3.0 technologies

Prerequisites...

This is an intermediate level programming course that also includes some advanced concepts. Students must have prior hands-on experience with the C# programming language and feel comfortable creating custom classes with members.



Course Outline...

Core C# and OOP Concepts

- Common Type System
 - Value Types
 - Reference Types
- Object Oriented Programming
 - Abstraction
 - Polymorphism
 - Inheritance
 - Encapsulation
 - Operator Overloading
 - Using Attributes

Foundation Class Library Overview

- Send E-mail
- Using System.Net Classes
- Pattern Matching with RegEx
- Efficient String Concatenation with the StringBuilder
- Using the StringReader/StringWriter Classes
- Dynamic Code Inspection with Reflection

Generics

- Why do we need Generics?
- Templates and Generics
- Generic Constraints
- Generics and Casting
- The default Keyword
- Nullable Types
- Using System.Collections.Generic Classes

Interfaces

- What are Interfaces?
- Creating Interfaces
- Implementing Interfaces
- Leveraging .NET Framework Interfaces

Events and Delegates

- What are Events, Delegates and Event Handlers?
- Creating Delegates and Events
- Raising and Handling Events



- Creating Custom EventArgs Classes
- Using the EventHandler<T> Delegate
- Anonymous Methods and Delegate Inference
- Asynchronous Operations with Delegates

Reading and Writing Files

- Using System.IO Classes
- Directory/DirectoryInfo
- File/FileInfo
- FileStream
- StreamReader
- StreamWriter
- FileSystemWatcher

Monitoring, Debugging and Instrumentation

- Using Debugging classes
- Tracing Operations
- Performance Counters

Serializing Data

- Binary serialization
- SOAP Serialization
- XML Serialization

Parsing XML Data

- Using System.Xml Classes
- XmlReader/XmlReaderSettings
- XmlWriter/XmlWriterSettings
- XmlDocument
- Creating Windows Services w/ FileSystemWatcher

Web Services

- Web Services Architecture
- Creating Services
- Consuming Services
- Asynchronous Web Service techniques

.NET Security

- Code Access Security
- Cryptography
- WindowsPrincipal/GenericPrincipal



ADO.NET

- ADO.NET Classes
- Generic Database Access with DbProviderFactories
- Multiple Active Resultsets (MARS)

Overview of .NET 3.0 Technologies

- Cardspace
- WPF
- WF
- WCF