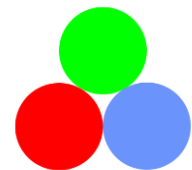


C#.Net Training Syllabus

C#/VB Language

Introducing C#/VB and the .NET Platform

- Understanding the Previous State of Affairs
- The .NET Solution
- Introducing the Building Blocks of the .NET Platform (the CLR, CTS, and CLS)
- Additional .NET-Aware Programming Languages
- An Overview of .NET Assemblies
- Understanding the Common Type System
- Understanding the Common Language Specification
- Understanding the Common Language Runtime
- The Assembly/Namespace/Type Distinction
- Exploring an Assembly Using ildasm.exe
- Exploring an Assembly Using Reflector
- Deploying the .NET Runtime
- The Platform-Independent Nature of .NET



The Philosophy of .NET

Building C#/VB Applications

- The Role of the .NET Framework 3.5/4.0 SDK
- Building C#/VB Applications Using csc.exe
- Building .NET Applications Using Notepad++
- Building .NET Applications Using Sharp Develop
- Building .NET Applications Using Visual Studio

Core C#/VB Programming Constructs

- The Anatomy of a Simple C#/VB Program
- An Interesting Aside: Some Additional Members of the System. Environment Class
- The System. Console Class
- System Data Types and C#/VB Shorthand Notation
- Working with String Data
- Narrowing and Widening Data Type Conversions
- Understanding Implicitly Typed Local Variables
- C#/VB Iteration Constructs

- Decision Constructs and the Relational/Equality Operators

Defining Encapsulated Class Types

- Introducing the C#/VB Class Type
- Understanding Constructors
- The Role of the this Keyword
- Understanding the static Keyword
- Defining the Pillars of OOP
- C#/VB Access Modifiers
- The First Pillar: C#/VB's Encapsulation Services
- Understanding Automatic Properties
- Understanding Object Initializer Syntax
- Working with Constant Field Data
- Understanding Partial Types

Understanding Inheritance and Polymorphism

- The Basic Mechanics of Inheritance
- Revising Visual Studio Class Diagrams

- The Second Pillar of OOP: The Details of Inheritance
- Programming for Containment/Delegation
- The Third Pillar of OOP: C#/VB's Polymorphic Support
- Understanding Base Class/Derived Class Casting Rules
- The Master Parent Class: System.Object

Understanding Structured Exception Handling

- The Role of .NET Exception Handling
- Configuring the State of an Exception
- System-Level Exceptions (System.SystemException)
- Application-Level Exceptions (System.ApplicationException)
- Processing Multiple Exceptions

Understanding Object Lifetime

- Classes, Objects, and References
- The Basics of Object Lifetime
- The Role of Application Roots
- Understanding Object Generations

- Concurrent Garbage Collection under .NET 1.0 – 4.0
- The System.GC Type
- Building Finalizable Objects
- Building Disposable Objects
- Building Finalizable and Disposable Types
- Understanding Lazy Object Instantiation

Advanced C#/VB Programming Constructs

Working with Interfaces

- Understanding Interface Types
- Implementing an Interface
- Interfaces As Parameters
- Interfaces As Return Values
- Arrays of Interface Types

Understanding Generics

- The Issues with Non-Generic Collections
- The Role of Generic Type Parameters
- The System.Collections.Generic Namespace

Delegates, Events, and Lambdas

- Understanding the .NET Delegate Type
- Defining a Delegate Type in C#
- The Simplest Possible Delegate Example
- Understanding C# Events
- Understanding Lambda Expressions

LINQ

- LINQ Specific Programming Constructs
- Understanding the Role of LINQ
- Returning the Result of a LINQ Query
- Investigating the C# LINQ Query Operators
- The Internal Representation of LINQ Query Statements

Programming with .NET Assemblies

Configuring .NET Assemblies

- The Role of .NET Assemblies
- Understanding the Format of a .NET Assembly
- Building and Consuming a Single-File Assembly

Introducing the .NET Base Class Libraries

Multithreaded and Parallel Programming

File I/O and Object Serialization