



.NET Framework using VB.NET (BDT4212) 4 days

Course Description

This four-day course is designed to provide a sound introduction to the .NET Framework for programmers who already know the Visual Basic language and the fundamentals of Windows Forms. It is current to .NET 3.5 and Visual Studio 2008. The course focuses on core portions of the .NET Framework that are common across many application areas. Separate courses are available in specific areas, such as ADO.NET, XML Programming, Windows Forms, Windows Presentation Framework, Windows Communications Framework, ASP.NET and Web services.

Learning Objectives

- Gain a thorough understanding of the philosophy and architecture of .NET
- Acquire a working knowledge of the .NET programming model and .NET Security
- Learn how to implement database applications using ADO.NET and LINQ
- Learn how to debug .NET applications using .NET diagnostic classes and tools

Course Duration: 4 days

Prerequisites: The student should be an experienced application developer or architect with a working knowledge of Visual Basic, including building simple GUIs with Windows Forms.

Course Outline

1. .NET Fundamentals
 - A. What is Microsoft .NET?
 - B. Common Language Runtime
 - C. Attribute-Based Programming
 - D. Interface-Based Programming
 - E. Metadata
 - F. Common Type System
 - G. Framework Class Library
 - H. Language Interoperability
 - I. Managed Code
 - J. Assemblies and Deployment
 - K. Web Services
 - L. ASP.NET
2. Class Libraries
 - A. Components in .NET
 - B. Building Class Libraries at the Command Line
 - C. Class Libraries Using Visual Studio 2008
 - D. Using References



.NET Framework using VB.NET (BDT4212) 4 days

3. Assemblies, Deployment and Configuration
 - A. Assemblies
 - B. Private Assembly Deployment
 - C. Shared Assembly Deployment
 - D. Configuration Overview
 - E. Configuration Files
 - F. Programmatic Access to Configuration
 - G. Using SDK Tools for Signing and Deployment
 - H. Application Settings with .NET 2.0

4. Metadata and Reflection
 - A. Metadata
 - B. Reflection
 - C. Late Binding

5. I/O and Serialization
 - A. Directories
 - B. Files
 - C. Serialization
 - D. Attributes

6. .NET Programming Model
 - A. Memory Management and Garbage Collection
 - B. Threading and Synchronization
 - C. Asynchronous Delegates
 - D. .NET 2.0 BackgroundWorker
 - E. Application Domains
 - F. Marshal by Value
 - G. Marshal by Reference
 - H. .NET Remoting

7. .NET Security
 - A. Authentication and Authorization
 - B. Configuring Security
 - C. Code Access Security
 - D. Code Groups
 - E. Evidence
 - F. Permissions
 - G. Role-Based Security
 - H. Principals and Identities



.NET Framework using VB.NET (BDT4212) 4 days

8. Interoperating with COM and Win32
 - A. .NET Client Calling a COM Server
 - B. PInvoke

9. ADO.NET and LINQ
 - A. ADO.NET Overview
 - B. .NET Data Providers
 - C. Connections
 - D. Commands
 - E. DataReaders and Connected Access
 - F. Data Sets and Disconnected Access
 - G. Language Integrated Query

10. Debugging Fundamentals
 - A. Compile-time Errors and Run-time Errors
 - B. Configuring Debug, Release, and Special Builds
 - C. Visual Studio 2008 Debugger
 - D. Just-In-Time Debugging

11. Tracing
 - A. Tracing
 - B. Event Logs

12. More About Tracing
 - A. Using the BooleanSwitch and TraceSwitch Classes
 - B. Print Debugging Information with the Debug Class
 - C. Instrumenting Release Builds with the Trace Class
 - D. Using Listeners
 - E. Implementing Custom Listeners

io-a/0