

Course ID  
**TK2415**

Course Duration  
**5 days**

Course Title

## **Programming with the Microsoft .NET Framework (Microsoft Visual Basic .NET)**

### **Aimed At**

Experienced, professional software developers including those employed by software companies or working on corporate development teams.

### **Group Size**

5-15

### **Prerequisites**

- Introduction to Visual Basic .NET Programming with Microsoft .NET (TK2559, 5 days)

### **Course In a Nutshell**

This course is designed to help application developers understand the Microsoft .NET Framework. In addition to offering an overview of the .NET Framework and an introduction to key concepts and terminology, the course provides a series of labs, which introduce and explain .NET Framework features that are used to code, debug, tune, and deploy applications. Covered are components, deployment, versioning, common type system, arrays, strings, collections, delegates, events, data streams and files, serialization, remoting, and XML web services.

### **Customize It!**

Ask us how you can customize this course to your own organization's requirements at little-to-no additional cost.

### **Learn How To**

- List the major elements of the .NET Framework and explain how they fit into the .NET platform
- Explain the main concepts behind the common language runtime and use the features of the .NET Framework to create a simple application
- Create and use components in Microsoft Windows forms-based and ASP.NET-based applications
- Use the deployment and versioning features of the .NET runtime to deploy multiple versions of a component
- Create, use and extend types by understanding the common type system architecture
- Create classes and interfaces that are functionally efficient and appropriate for given programming scenarios
- Use the .NET Framework class library to efficiently create and manage strings, arrays, collections and enumerators
- Use delegates and events to make an event sender signal the occurrence of an action to an event receiver
- Describe and control how memory and other resources are managed in the .NET Framework
- Read from and write to data streams and files
- Use the basic request/response model to send and receive data over the Internet
- Serialize and deserialize an object graph

- Create distributed applications using XML web services and object remoting

## Course Outline

- Overview of Microsoft .NET Framework
  - Describing the .NET Framework and its components
  - Overview of namespaces
  - Relationship between the .NET Framework class library and namespaces
- Introduction to Managed Execution Environment
  - Writing a .NET Framework application
  - Compiling and running a .NET Framework application
  - Concept of garbage collection
- Working with Components
  - Introduction to key .NET Framework development technologies
  - Creating a simple .NET Framework component
  - Implementing structured exception handling
  - Creating a simple console client
  - Creating an ASP.NET client
- Deployment and Versioning
  - Introduction to application deployment
  - Packaging and deploying simple and componentized applications
  - Application deployment scenarios
  - Installing and removing assemblies from the global assembly cache
  - Related topics and tools
- Common Type Systems
  - Introduction to the common type system
  - Describing the difference between value types and reference types
  - Elements of the common type system
  - Object-oriented characteristics
- Working with Types
  - System.Object class functionality
  - Specialized constructors
  - Type operations
  - Interfaces
  - Managing external types
- String, Arrays and Collections
  - Parsing, formatting, manipulating and comparing strings
  - Terminology of collections
  - .NET Framework arrays
  - .NET Framework collections
- Delegates and Events
  - Explanation and use of delegates

- Explanation and use of multicast delegates
- Explanation of events
- Implementing events that conform to .NET Framework guidelines
- Memory and Resource Management
  - Basics of memory management
  - Management of non-memory resources
  - Managing implicit resources
  - Managing explicit resources
  - Optimizing garbage collection
- Data Streams and Files
  - Using stream objects
  - Using readers and writers
  - Basic file I/O
  - Explanation of the key features of the .NET Framework's isolated storage mechanism
- Internet Access
  - Internet application scenarios
  - The WebRequest and WebResponse model
  - Application protocols
  - Handling errors
  - Security
  - Best practices
- Serialization
  - Serialization scenarios
  - Serialization attributes
  - Object graph
  - Serialization process
  - Serialization example
  - Deserialization example
  - Custom serialization
  - Custom serialization example
  - Security issues
- Remoting and XML Web Services
  - Writing and configuring distributed applications that use .NET remoting
  - Creating an XML web service by using Visual Studio .NET and ASP.NET
  - Consuming an XML web service by using the web services description language tool (Wsdll.exe)
- Wrap-up: Course Recap, Q/A, and Evaluations

### **How You Will Learn**

- An experienced IT professional and instructor will teach this class as a hands-on workshop.
- In-class labs and exercises will help you thoroughly understand each topic.
- The instructor will offer real-life examples, applications, and case studies to

help you make a bridge from the theory to its real-world application.

- You'll learn the "secrets" known to the professionals, including the dos and don'ts, tricks, and hidden pitfalls.
- You will receive a comprehensive student manual that you can use as an on-the-job refresher and reference.

*Revised*

*April 6, 2007*