

Introduction to Microsoft .NET Development

Course 2717—Two days—Instructor-led

Introduction

This two-day instructor-led seminar provides students with an introduction to the technologies that comprise the Microsoft .NET strategy. This course is a high-level overview of multiple aspects of .NET, and is intended as a starting point for developers and business decision-makers to evaluate Microsoft .NET tools and technologies. As an introductory seminar, it provides pointers to training courses and other resources that give more detail on specific topics.

Audience

This course is intended for experienced programmers who are evaluating Microsoft .NET tools and technologies, such as Microsoft Visual Studio .NET 2003 and the .NET Framework 1.1, and provides pointers to resources where they can obtain specific skills for creating applications, components, and services built on the .NET Framework.

At Course Completion

After completing this course, students will have a fundamental understanding of how to:

- Create a simple .NET solution by using Microsoft Visual J# .NET, Microsoft Visual C++ .NET, and Microsoft Visual Basic .NET.
- Use Microsoft Visual Studio .NET.
- Build simple ASP.NET applications.
- Access data in a variety of formats.
- Apply object-oriented programming concepts.
- Create a simple .NET application based on the Microsoft Windows application template.
- Use the Extensible Markup Language (XML).
- Create an XML Web service.
- Build scalable and reliable applications.
- Apply core security concepts in the .NET Framework 1.1.
- Use the lifecycle tools included in Visual Studio .NET.
- Configure and deploy an application built on the .NET Framework 1.1.
- Interoperate with existing applications.
- Migrate existing applications to the .NET Framework 1.1.
- Build applications for mobile devices.
- Develop applications for Windows Server 2003.

Prerequisites

Before attending this course, students should have:

- Basic programming experience with Microsoft Visual Basic, Microsoft Visual Basic Scripting Edition, C, C++, or Java.
- Familiarity with Microsoft development technologies such as Windows Forms and Active Server Pages (ASP).

Microsoft Certified Professional Exams

No Microsoft Certified Professional exams are associated with this course currently.

Course Materials

The student kit includes a comprehensive workbook and other necessary materials for this class.

Course Outline

Module 1: Introduction to Microsoft .NET

This module provides students with an introduction to Microsoft .NET. The module identifies problems with existing technologies that are solved by the .NET initiative, and describes the core services and features provided by the .NET Framework 1.1. After completing this module, students will be able to describe the features and benefits of using XML Web services and the .NET Framework 1.1.

Lessons

- The Microsoft Application Platform
- Introduction to XML Web Services
- Inside the .NET Framework
- Introduction to .NET Languages
- Microsoft Developer Tools Roadmap

After completing this module, students will be able to:

- Describe how the Microsoft Application Platform simplifies development.
- Identify the features and benefits of XML Web services.
- Describe the core components of the .NET Framework.
- Describe how multiple languages are supported by the .NET Framework.
- Describe the Microsoft Developer Tools Roadmap.
- Describe the features and benefits of Microsoft .NET.

Module 2: Using Visual Studio .NET

This module shows how Microsoft Visual Studio .NET is a suite of programming tools that provide a complete development environment for building applications for the Microsoft .NET platform. After completing this module, students will be able to describe the design goals, features, and benefits of using Microsoft Visual Studio .NET.

Lessons

- Introduction to Visual Studio .NET
- Using Visual Studio .NET for Web Application Development

After completing this module, students will be able to:

- Explain the design goals of Visual Studio .NET.
- List the productivity features of Visual Studio .NET.
- Describe the Visual Studio .NET Rapid Application Development (RAD) tools.
- Understand how Visual Studio .NET simplifies the Web application development process and shortens delivery time.

Module 3: Building ASP .NET Applications

This module describes Microsoft Active Server Pages (ASP) as they are implemented for the .NET platform. After completing this module, students will be able to describe how ASP.NET can be used to develop enterprise-class Web applications, including those designed for mobile browsers.

Lessons

- Introduction to ASP .NET
- Using Web Forms
- Introduction to ASP.NET Mobile Web Applications
- Introduction to ASP.NET Application Services

After completing this module, students will be able to:

- Discuss the capabilities of ASP.NET and its implementation.
- Identify the important concepts of Web Forms and ASP.NET server controls.
- Discuss the capabilities of ASP.NET Mobile Web Applications.
- Describe the features of the ASP.NET application services.

Module 4: Using ADO.NET

This module describes a new data access technology that is an evolutionary improvement to Microsoft ActiveX Data Objects (ADO). ADO.NET is a group of classes in the Microsoft .NET Framework that integrates previous XML and ADO object models, and is designed for distributed applications that might need to operate over the Web. After completing this module, you will be able to describe how to implement data services across enterprise-level applications using ADO.NET.

Lessons

- Introduction to ADO.NET
- The ADO.NET Object Model
- Using .NET Framework Data Providers

After completing this module, students will be able to:

- Describe ADO.NET and its architecture.
- Use the ADO.NET and DataSet objects.
- Use the .NET Framework data providers.

Module 5: Applying Object-Oriented Programming Concepts

This module describes how object-oriented programming techniques are available across .NET languages, and how the .NET common language runtime (CLR) provides features to simplify the development of reusable components. After completing this module, you will be able to define object-oriented concepts such as inheritance and describe how namespaces and assemblies can be used to increase efficiency in component development.

Lessons

- Introduction to Classes and Their Members
- Inheritance
- Namespaces and Assemblies

After completing this module, students will be able to:

- Define a class and identify its members.
- Identify how to implement inheritance in .NET.
- Describe how .NET relies on namespaces and assemblies.

- Describe how object-oriented programming concepts apply to .NET development.

Module 6: Developing Windows Applications

This module explains how to use existing forms and controls that are available for an application's user interface. After completing this module, you will be able to create forms and form templates, and describe how to utilize some advanced functionality of existing controls.

Lessons

- Introduction to Windows Applications
- Using Windows Forms
- Visual Inheritance

After completing this module, students will be able to:

- Describe the features of Windows Forms.
- Create a Windows Forms application.
- Create a form that inherits from another form.

Module 7: Using XML in .NET

This module explains how you can work with XML by using classes defined within the .NET Framework. After completing this module, you will be able to explain the role of XML classes in the .NET Framework, describe how ADO.NET takes advantage of the power of XML to provide disconnected access to data, and retrieve and write XML data. You will also use the XML Designer to create and edit XML source code and XML Schema Definition (XSD) schemas.

Lessons

- XML in .NET
- Using the XML Designer
- ADO.NET and XML

After completing this module, students will be able to:

- Explain the role of XML classes in the .NET Framework.
- Describe how Microsoft ActiveX Data Objects .NET (ADO.NET) takes advantage of the power of XML to provide disconnected access to data.
- Retrieve and write XML data.
- Use the XML Designer to create and edit XML source code and XML Schema Definition (XSD) schemas.

Module 8: XML Web Services

This module describes a simple, standards-based model for binding applications together over the Internet by using XML Web services. After completing this module, you will be able to describe how to create, implement, deploy, secure, and consume an XML Web service.

Lessons

- Introduction to XML Web Services
- Creating and Implementing XML Web Services
- Deploying and Securing XML Web Services
- Consuming XML Web Services

After completing this module, students will be able to:

- Identify XML Web services.
- Design an XML Web service.
- Create and implement an XML Web service.

- Deploy and secure an XML Web service.
- Access an XML Web service from a client application.

Module 9: Making Applications Scalable and Reusable

This module describes the tools built into the .NET Framework for creating reliable and scalable applications. After completing this module, you will be able to explain Microsoft Message Queuing support in .NET. You will also be able to describe how Microsoft Visual Studio .NET Server Explorer and programmatic access to event logs, performance counters, and tracing can help you to deploy and test your applications.

Lessons

- Using Microsoft Message Queuing
- Application Diagnostics and Tuning

After completing this module, students will be able to:

- Use Microsoft Message Queuing to build scalable applications.
- Diagnose and tune applications by using event logs, performance counters, tracing, and Microsoft Windows Services.
- Describe the following testing tools: Microsoft Visual Studio Analyzer and Application Center Test.

Module 10: Security in .NET

This module describes the core security features provided by the .NET Framework. After completing this module, you will be able to explain how to implement authorization and authentication for applications created within the framework. You will also be able to identify additional security measures and tools provided by the framework.

Lessons

- Code-Based Security
- Role-Based Security
- Additional Security Measures

After completing this module, students will be able to:

- Understand code-based security.
- Understand role-based security.
- Implement the security mechanisms provided by the .NET Framework.
- Understand how the .NET Framework handles authorization and authentication.
- Identify the additional security measures and security tools provided by the .NET Framework.

Module 11: Lifecycle Tools for Microsoft .NET

This module describes Microsoft Visual Studio .NET tools that help you to design, develop, and deploy enterprise-level applications. After completing this module, you will be able to describe the features of enterprise templates and describe how the Unified Modelling Language (UML) and Object Role Modelling (ORM) are applied in software design. You will also be able to describe the use of Application Centre Test and Microsoft Visual SourceSafe to simplify project deployment and testing.

Lessons

- Using Enterprise Templates
- Software Modelling Using UML
- Data Modelling

- Using Application Centre Test
- Introduction to Visual Source Safe

After completing this module, students will be able to:

- Describe the features, goals, and components of Enterprise Templates.
- Discuss software modelling by using a Unified Modelling Language (UML).
- Explain database modelling by using Object Role Modelling (ORM).
- Describe how to use Application Centre Test.
- Discuss the features of Microsoft Visual SourceSafe.
- Describe the lifecycle tools included with Visual Studio .NET 2003 Enterprise Architect Edition

Module 12: Configuring and Deploying .NET-based Applications

This module describes how the .NET Framework and the CLR help you to configure and deploy self-described, self-contained applications. After completing this module, you will be able to describe how to use assemblies and the global assembly cache to configure and deploy applications.

Lessons

- Configuring .NET Framework Applications
- Using Assemblies in .NET Framework Applications
- Deploying .NET Framework Applications

After completing this module, students will be able to:

- Identify important concepts of configuring .NET Framework applications.
- Identify the role of assemblies and the global assembly cache in configuring and deploying .NET applications.
- Deploy .NET Framework applications.
- Describe how to configure and deploy .NET assemblies.

Module 13: Application Interoperability

This module explains how Microsoft Visual Studio .NET helps you to enhance the reliability of your applications by using managed code. After completing this module, you will be able to call COM components from the .NET Framework, and call .NET Framework components from COM.

Lessons

- Introduction to Application Interoperability
- Calling COM Objects from .NET
- Calling .NET Components from COM
- Using Platform Invoke

After completing this module, students will be able to:

- Describe the role of interop services with reference to the Microsoft .NET Framework.
- Call COM components from the .NET Framework.
- Call the .NET Framework components from COM.
- Describe how to use Platform Invocation Services (PInvoke) to call unmanaged functions implemented in DLLs.

Module 14: Migrating Applications to .NET

This module explains how to migrate legacy applications to the .NET Framework by using the Upgrade Wizard. After completing this module, you will be able to port applications from earlier

versions of Visual Basic and ASP to Visual Basic .NET and ASP.NET. You will also learn how to convert Microsoft Visual J++ 6.0 projects to J# or C# code.

Lessons

- Migrating Visual Basic Applications to .NET
- Migrating ASP Applications to ASP.NET
- Migrating Visual J++ Applications to .NET

After completing this module, students will be able to:

- Migrate Visual Basic 6.0 applications to Visual Basic .NET.
- Migrate ASP applications to ASP.NET.
- Migrate Java-language code to .NET.

Module 15: Building Applications for Mobile Devices

This module describes how Microsoft Visual Studio .NET 2003 is a complete set of development tools for building Smart Device applications that run on mobile devices. After completing this module, you will be able to describe the .NET Compact Framework, create Smart Device applications, and use SQL Server CE to store data locally.

Lessons

- Introduction to the .NET Compact Framework
- Creating Smart Device Applications
- Using SQL Server CE To Store Data

After completing this module, students will be able to:

- Describe the .NET Compact Framework.
- Create Smart Device applications.
- Use Microsoft SQL Server™ 2000 Windows CE to store data locally.

Module 16: Developing with Windows Server 2003

This module describes Windows Server 2003 as a .NET development environment. In this module, you will see how Windows Server 2003 builds on the core strengths of the Windows family of operating systems - security, manageability, reliability, availability, and scalability. After completing this module, you will be able to describe the Web technologies and application server features provided by Windows Server 2003, and be able to discuss the benefits of developing with Windows Server 2003.

Lessons

- Using Windows Server 2003 As An Application Server
- Using Windows Server 2003 Web Technologies

After completing this module, students will be able to:

- Describe the application server features of Windows Server 2003.
- Describe the Web technologies of Windows Server 2003.
- Describe the benefits of developing with Windows Server 2003.