MOC 2548 Core Distributed Application Development with Microsoft Visual Studio 2005

Course Summary

Description

This workshop provides students with the knowledge and skills to develop distributed applications by using the Microsoft .NET Framework and Microsoft Visual Studio 2005. The workshop focuses on building distributed applications by using Web services, remoting, Microsoft Message Queuing, and serviced components.

Objectives

At the end of this course, students will be able to:

- Build and use a Web service.
- Configure and customize a Web service application.
- Call Web methods asynchronously.
- Build remote client and server applications.
- Create and serialize remoteable types.

- Manage the lifetime of remote objects.
- Call remote methods asynchronously.
- Implement remote events.
- Send and receive messages by using Microsoft Message Queuing.
- Create and use serviced components.

Topics

- Building and Consuming a Simple XML Web Service
- Configuring and Customizing a Web Service
- Calling Web Methods Asynchronously
- Building a Remoting Client and Server
- Creating and Serializing Remotable Types

- Performing Remoting Operations Asynchronously
- Managing the Lifetime of Remote Objects
- Sending and Receiving Messages by Using Message Queuing
- Creating and Consuming Serviced Components

Audience

This workshop is intended for corporate and Independent software vendor application developers who have a desire to learn more about specific technology areas in distributed application development.

Prerequisites

Before attending this workshop, students must:

- Be able to manage a solution environment using the Visual Studio 2005 Integrated development environment (IDE) and tools
- Understand the Microsoft .NET Framework 2.0 and the Common Language Runtime
- Be able to program an application by using a .NET Framework 2.0-compliant language
- Know how to make assemblies available to other applications
- Have a basic understanding of XML including XML declaration, elements, attributes, and namespaces
- Have a basic understanding of application domains
- Have a basic understanding of delegates and events
- Have a basic understanding of threads

Duration

Three days

MOC 2548 Core Distributed Application Development with Microsoft Visual Studio 2005

Course Outline

I. Building and Consuming a Simple XML Web Service

This unit describes how you can create a simple Web service and client application by using the .NET Framework. It also explains how you can configure client proxies, and debug and deploy Web services.

   A. Technical Context of Web Services
   B. Components of Web Service Technology

Lab: Building and Consuming a Simple Web Services

- Exercise 1. Creating a Web Service and Client
- Exercise 2. Working with the Client Proxy
- Exercise 3. Deploying a Web Service and Configuring a Client
- Exercise 4. Debugging and Exception Handling in Web Services
- Exercise 5. Determining Web Service Connectivity

After completing this module, students will be able to:

- Explain the technical context for Web services.
- Understand key components of Web service technology.
- Create a Web service and client.
- Configure a Web service client and proxy.
- Deploy and use a Web service.
- Debug a Web service.
- Determine Web service connectivity.

II. Configuring and Customizing a Web Service

This unit introduces a number of important configuration and customization options for Web services. It describes how to control the way in which complex parameters to Web methods are serialized. This unit also shows how to use configuration files to control the way in which a Web service operates.

   A. XML Serialization
   B. How to Use Complex Data Types in Web Services
   C. How to Use Attributes to Control Serialization
   D. How to Use Service Configuration Attributes
   E. Configuration Files

Lab: Configuring and Customizing a Web Service

- Exercise 1. Creating and Using Custom Data Types
- Exercise 2. Customizing the Web Service

- Exercise 3. Configuring the Web Service Using the Web.config File

After completing this unit, students will be able to:

- Explain XML serialization of user-defined classes.
- Pass complex data types between a Web service and a client.
- Configure XML serialization.
- Configure SOAP formatting options.
- Configure the namespace and binding for a Web service.
- Configure a Web service by using the Web.config file.

III. Calling Web Methods Asynchronously

This unit explains how to call a Web method asynchronously. It describes how to improve the responsiveness of client applications by avoiding the need to wait for Web methods to complete execution before continuing processing. This unit covers the different options available for calling Web methods asynchronously and it describes how to create one-way methods.

   A. The Need for Asynchronous Calls
   B. Options for Making Asynchronous Calls
   C. One-Way Methods

Lab: Calling Web Methods Asynchronously

- Exercise 1. Using a One-Way Method
- Exercise 2. Calling a Web Method Asynchronously

After completing this unit, students will be able to:

- Explain why asynchronous calls are needed by Web service clients.
- Create and invoke one-way methods.
- Call methods in a Web service asynchronously.

IV. Building a Remoting Client and Server

This unit describes key remoting concepts, and shows how to create a remoting server and client. This unit describes how to use remoting to call methods in remote objects, and how to pass data across remoting boundaries. This unit also shows how to configure and deploy remoting applications.

   A. Technical Context of Remoting
   B. Remoting Servers and Clients
   C. Important Components of Remoting

MOC 2548 Core Distributed Application Development with Microsoft Visual Studio 2005

Course Outline (cont'd)

Lab: Building a Remoting Client and Server
- Exercise 1. Implementing a Simple Remoting Client and Server
- Exercise 2. Passing Data by Value
- Exercise 3. Configuring Remoting Channels and Activation Modes Programmatically
- Exercise 4. Configuring Remoting Channels and Activation Modes with Configuration Files
- Exercise 5. Deploying and Debugging Remotable Classes

After completing this unit, students will be able to:
- Describe the technical context of remoting.
- Implement a simple remoting server and client.
- Pass data by value across a remoting boundary.
- Configure remoting channels.
- Use different activation modes.
- Configure a remoting service by using a configuration file.
- Deploy and host remotable types.
- Debug a remotable type.

V. Creating and Serializing Remotable Types

This unit describes how to transfer complex data values across remoting boundaries, and the issues involved in doing so. It compares and contrasts the marshal by value and marshal by reference mechanisms for accessing remote data. This unit also covers version compatibility issues between clients and servers using different versions of a class, and the special requirements for remoting generic classes.
- A. Marshal by Value
- B. Marshal by Reference
- C. Version Compatibility for Remotable Types
- D. Generic Classes

Lab: Creating and Serializing Remotable Types
- Exercise 1. Using Serialization Formatters
- Exercise 2. Using Marshal by Reference
- Exercise 3. Using Version Tolerant Serialization

After completing this unit, students will be able to:
- Describe the differences between marshal by value and marshal by reference.
- Describe the issues surrounding versioning and remoting.
- Use version tolerant serialization.

- Configure a communication channel to use different serialization formatters.
- Create and marshal remotable objects by reference.

VI. Performing Remoting Operations Asynchronously

This unit describes how to call a method asynchronously in the remoting environment. It covers the different techniques you can use and it explains how to raise events in a remoting server and handle them in a client.
- A. Asynchronous Methods
- B. Calling Remote Methods Asynchronously
- C. One-Way Methods
- D. Using Events in Remoting Applications

Lab: Performing Remoting Operations Asynchronously
- Exercise 1: Calling Remote Methods Asynchronously
- Exercise 2: Raising and Handling Events in Remoting

After completing this unit, students will be able to:
- Call remoting methods asynchronously by using BeginInvoke.
- Implement callbacks.
- Create and call one-way methods.
- Create and fire events in remote services.
- Handle events in a client application.

VII. Managing the Lifetime of Remote Objects

This unit describes the lifetime of remote objects and how you can control them. This unit introduces the concepts of remote object leases and sponsors. This unit shows how to initialize a remote object's lease to a specific period, and how to renew an object's lease when it expires by using a sponsor.
- A. Life Cycle of Remote Objects
- B. Lifetime Sponsors
- C. Lease Properties
- D. Leases and Exception Handling

Lab: Managing the Lifetime of Remote Objects
- Exercise 1: Initializing the Lifetime of Remote Objects
- Exercise 2: Renewing the Lifetime of Remote Objects

MOC 2548 Core Distributed Application Development with Microsoft Visual Studio 2005

Course Outline (cont'd)

After completing this unit, students will be able to:
- Initialize the lifetime of a remote object.
- Renew the lifetime of a remote object.
- Configure the renewal properties of a lease.
- Handle exceptions caused by lease expiry.

VIII. Sending and Receiving Messages by Using Message Queuing

This unit describes how to use Microsoft Message Queuing to build distributed applications. It covers the essential aspects of building client and server applications that use message queues, how to create queues, how to send and receive messages, and how to handle replies to messages. This unit also describes how to access message queues across the Internet.
- A. Understanding Message Queuing
- B. Creating a Message Queue and Sending a Message
- C. Receiving a Message and Posting a Response
- D. Using IIS with Message Queuing

Lab: Sending and Receiving Messages by Using Message Queuing
- Exercise 1. Building a Simple Messaging Client and Server
- Exercise 2. Using More Complex Data Types and Formatters
- Exercise 3. Using Response Queues and Time-Outs
- Exercise 4. Placing Messages on a Queue by Using IIS and HTTP

After completing this unit, students will be able to:
- Explain message queuing.
- Create a message queue and send messages.
- Receive a message and post a response.
- Use Internet Information Services with message queuing.

IX. Creating and Consuming Serviced Components

This unit explains how to build and access serviced components in a .NET Framework application. This unit describes the relationship between .NET Framework serviced components and COM+. It shows how to use the .NET Framework to implement a serviced component that you can register as a COM+ application and how you can write applications that use serviced components.
- A. COM+ Services
- B. Implementing a Serviced Component
- C. Registering a Serviced Component
- D. Instantiating a Serviced Component

Lab: Creating and Consuming Serviced Components
- Exercise 1. Creating and Using a Serviced Component
- Exercise 2. Using Enterprise Services in a Serviced Component

After completing this unit, students will be able to:
- Understand the role of COM+ services.
- Implement a serviced component.
- Register a serviced component.
- Instantiate a serviced component.