



To register or for more information call our office **(208) 898-9036** or email register@leapfoxlearning.com

Microsoft®

MS-2546 Intense Windows Client Development with Visual Studio 2005

Duration:

Traditional Instructor Led Learning -5 days

Software Assurance Vouchers Accepted



Introduction

This five-day instructor-led workshop provides students with the knowledge and skills to develop Microsoft Windows Forms applications using Microsoft Visual Studio 2005. The workshop focuses on user interfaces, program structure, and implementation details.

Audience

This workshop is intended for application developers who have a desire to learn more about specific technology areas in Windows application development.

At Workshop Completion

After completing this workshop, students will be able to:

- Create a simple Windows Forms application.
- Configure standard controls.
- Build menus.
- Display and edit data by using data-bound controls.
- Provide user assistance and enhance usability.
- Create consistent applications by using dialogs and forms inheritance.
- Add print and report functionality to a Windows Forms application.
- Perform asynchronous tasks by using the BackgroundWorker component.
- Deploy a Windows Forms application by using ClickOnce.
- Build MDI applications.
- Customize Windows Forms and controls.
- Create customized print components.
- Perform drag-and-drop operations and implement Clipboard support.
- Perform asynchronous tasks in Windows Forms by using multithreaded techniques.
- Enhance the presentation of Windows Forms applications.

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 Microsoft .NET Framework 2.0 and the Common Language Runtime
- Be able to program an application 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.

Workshop Outline

Unit 1: Creating a Simple Windows Forms Application

This unit introduces the fundamental skills required to create a simple Windows Forms application. It explains how to configure form properties and how to add controls to a form. It also deals with events and explains how to create event handlers at design time and run time.
Lessons

- Components of a Windows Forms User Interface
- Event Handling in a Windows Forms Application

Lab 1: Creating a Simple Windows Forms Application

- Exercise 1. Adding and Configuring Windows Forms and Controls
- Exercise 2. Creating Event Handlers and Monitoring Events
- Exercise 3. Adding Controls and Handlers at Run Time
- Exercise 4. Adding a Form Keyboard Handler (if time permits)

After completing this unit, students will be able to:

- Describe the various components of a Windows Forms application.
- Explain how to handle events in a Windows Forms application.
- Add and configure Windows Forms and controls.
- Create event handlers and monitor events.
- Add controls and handlers to a form at run time.
- Apply best practices when designing forms.

Unit 2: Configuring Standard Controls

This unit introduces many of the controls from the Visual Studio Toolbox. It teaches how to add and configure these controls and explains how to group them into different categories by function.

Lesson

- Windows Forms Controls by Function

Lab 2: Configuring Standard Controls

- Exercise 1. Creating the EmployeeViewer Form
- Exercise 2. Creating the ReportGenerator Form
- Exercise 3. Creating the ReportGeneratorHelp Form

After completing this unit, students will be able to:

- Explain the functions of the major categories of Windows Forms controls.
- Create a form to select and view images by using the SplitContainer, TreeView, PictureBox, ToolStrip, and other controls.
- Create a form to specify report criteria by using the TableLayoutPanel, ListView, MonthCalendar, ComboBox, and other controls.

- Create a form to view a Hypertext Markup Language (HTML) Help file by using a WebBrowser control.

Unit 3: Building Menus

This unit introduces the MenuStrip control and the ContextMenuStrip component. It explains how to create and configure form menus and context menus in an application. It also deals with the ToolStripItems that can be added to the container of a MenuStrip or ContextMenuStrip.

Lesson

- Menus in Windows Forms

Lab 3: Building Menus

- Exercise 1. Creating and Configuring a MenuStrip Control
- Exercise 2. Creating and Configuring a ContextMenuStrip Component
- Exercise 3. Hiding and Disabling a ToolStripMenuItem
- Exercise 4. Creating and Configuring a ToolStripComboBox (if time permits)

After completing this unit, students will be able to:

- Explain the structure of a menu in a Windows Forms application.
- Create and configure a MenuStrip control.
- Create and configure a ContextMenuStrip component.
- Hide and disable a ToolStripMenuItem.
- Apply best practices for menu design and development.

Unit 4: Displaying and Editing Data by Using Data-Bound Controls

This unit introduces the controls that can be used to display data from a data source. It shows how to use Visual Studio 2005 to create data sources and add data-bound controls to a form. It also demonstrates how to use the DataGridView control to display and update data retrieved by using a data source.

Lessons

- Binding Data to a Control
- DataGridView Control

Lab 4: Displaying and Editing Data by Using Data-Bound Controls

- Exercise 1. Adding a Data Source to a Project
- Exercise 2. Binding Data to a Control
- Exercise 3. Creating a Bound Form by Using the BindingSource and BindingNavigator Components
- Exercise 4. Displaying Data by Using the DataGridView Control
- Exercise 5. Binding to Data at Run Time (if time permits)

After completing this unit, students will be able to:

- Explain simple and complex data binding.
- Describe the scenarios in which the DataGridView control is used.
- Add a data source to a project.

- Bind data to a control at run time.
- Use the BindingSource component and BindingNavigator control to create a data-bound Windows Form.
- Use the DataGridView control to display and update tabular data from a data source.

Unit 5: Providing User Assistance and Enhancing Usability

This unit introduces many of the controls and techniques that can be used to create an application that is flexible and intuitive and that provides timely feedback to the user. It shows how to add and configure the available user assistance controls to provide ToolTips, Help, and information about errors. It also describes the accessibility features of Windows Forms and explains how to implement globalization and localization in an application.

Lessons

- Providing User Assistance
- Implementing Accessibility Features
- Implementing Globalization and Localization

Lab 5: Providing User Assistance and Enhancing Usability

- Exercise 1. Adding and Configuring User Assistance Controls
- Exercise 2. Implementing Accessibility Features
- Exercise 3. Implementing Globalization and Localization
- Exercise 4. Running a Procedure at Set Intervals (if time permits)
- Exercise 5. Displaying Data Source Errors (if time permits)

After completing this unit, students will be able to:

- Describe the components that can be used to provide user assistance.
- Add and configure user assistance controls on a Windows Form.
- Explain the accessibility features available to implement in a Windows Forms application.
- Implement accessibility features within a Windows Forms application.
- Explain the high-level processes involved in globalization and localization in a Windows Forms application.
- Implement globalization and localization in a Windows Form.

Unit 6: Creating Consistent Applications by Using Dialog Boxes and Forms Inheritance

This unit introduces the built-in dialog boxes that can be used to prompt users when they are performing common tasks and to provide users with a familiar interface. It explains how to add and configure dialog boxes that enable users to open and save files and to set font and color properties. This unit also explains how to create and use a custom dialog box. In addition, this unit explains the concept of forms inheritance and describes how to create a consistent interface for Windows Forms applications.

Lessons

- Dialog Boxes in a Windows Forms Application
- Windows Forms Inheritance

Lab 6: Creating Consistent Applications by Using Dialog Boxes and Forms Inheritance

- Exercise 1. Adding and Configuring Dialog Box Components
- Exercise 2. Creating a Custom Dialog Box

- Exercise 3. Implementing Windows Forms Inheritance
- Exercise 4. Modifying an Existing Form to Inherit from a Base Form (if time permits)

After completing this unit, students will be able to:

- Describe the various standard preconfigured dialog boxes available to a Windows Forms application.
- Explain the concept of Windows Forms inheritance and the reasons for using inheritance.
- Add and configure dialog box controls on a Windows Form.
- Create a custom dialog box.
- Implement Windows Forms inheritance.

Unit 7: Printing Content and Creating Reports

This unit provides an introduction to the components that can be used to preview and print reports from a Windows Forms application. This unit covers the predefined dialog boxes that simplify the processes involved, and it explains how to use these dialog boxes to retrieve print settings and page setup options from the user.

In addition, this unit explains how to display a report in a Windows Forms application by using the CrystalReportViewer component

Lessons

- Printing in a Windows Forms Application
- Reporting in a Windows Forms Application

Lab 7: Printing Content and Creating Reports

- Exercise 1. Adding Print Capability to a Windows Forms Application
- Exercise 2. Implementing Reporting by Using the CrystalReportViewer Component

After completing this unit, students will be able to:

- Explain how to print a report in a Windows Forms application.
- Describe how to create and view reports in a Windows Forms application.
- Use the components provided with Visual Studio 2005 to add print capability to a Windows Forms application.
- Display a Crystal report by using the CrystalReportViewer component.

Unit 8: Performing Asynchronous Tasks by Using the BackgroundWorker Component

This unit introduces the main concepts of asynchronous programming and then focuses on the BackgroundWorker component. It explains how to work with the methods and events of the BackgroundWorker component to add asynchronous functionality to a Windows Forms application.

Lesson

- Asynchronous Tasks in Windows Forms Applications

Lab 8: Performing Asynchronous Tasks by Using the BackgroundWorker Component

- Exercise 1. Running and Canceling a Background Process by Using the BackgroundWorker Component

After completing this unit, students will be able to:

- Explain the concepts and processes involved in asynchronous programming.
- Run and cancel a background process by using the BackgroundWorker component.
- Describe best practices for using the BackgroundWorker component.

Unit 9: Deploying Applications by Using ClickOnce

This unit explains how to deploy a Windows Forms application by using ClickOnce. It covers the steps required to prepare, publish, install, and test an application. Finally, this unit explains how to update an application and how to use the automatic update feature of ClickOnce.

Lessons

- Windows Forms Application Deployment Options
- ClickOnce Technology Overview

Lab 9: Deploying Applications by Using ClickOnce

- Exercise 1. Publishing a ClickOnce Application
- Exercise 2. Installing and Testing a ClickOnce Application

After completing this unit, students will be able to:

- Explain the options for deploying a Windows Forms application.
- Explain the key concepts of ClickOnce technology.
- Publish a ClickOnce application.
- Install and test a ClickOnce application.
- Describe best practices for using ClickOnce deployment.

Unit 10: Building MDI Applications

This unit explains how to create multiple-document interface (MDI) applications that enable one parent window to host multiple documents. It demonstrates how to create MDI parent and child forms and how to determine the active MDI child and work with information on it. It also explains how to implement menu merging in an MDI application to make the menu on the parent form relevant to the active child form.

Lessons

- Windows Forms Layout Options.
- What Are MDI Applications?

Lab 10: Building MDI Applications

- Exercise 1. Creating MDI Parent and Child Forms.
- Exercise 2. Displaying and Comparing Information on MDI Child Forms.
- Exercise 3. Implementing Menu Merging in MDI Applications.

After completing this unit, students will be able to:

- Compare the different layout styles for Microsoft Windows Forms applications.
- Explain the key concepts and processes involved in implementing an MDI application.
- Create MDI parent and child forms.

- Display and compare information on MDI child forms.
- Implement menu merging in an MDI application.

Unit 11: Customizing Windows Forms and Controls

This unit explains how to develop custom Microsoft Windows Forms and controls. Students will learn how to develop user controls, use GDI+ operations, and create new controls that inherit from the Control class. In addition, it demonstrates how to create a nonrectangular Windows Form and how to add features such as attributes and Toolbox bitmaps to controls.

Lesson

- What Are the Methods of Authoring Controls for Windows Forms?
- Ways to Draw a User Interface by Using GDI+.
- Creating a Nonrectangular Windows Form.

Lab 11: Customizing Windows Forms and Controls

- Exercise 1. Creating a Control that Inherits from an Existing Control.
- Exercise 2. Creating a Nonrectangular Windows Form.
- Exercise 3. Creating a Custom User Control.

After completing this unit, students will be able to:

- Explain the methods of authoring controls for Windows Forms.
- Draw a user interface by using GDI+.
- Create a control that inherits from an existing control.
- Create a nonrectangular Windows Form.
- Create a custom user control.

Unit 12: Creating Customized Print Components

This unit explains how to print content from a Microsoft Windows Forms application by using the printing features of GDI+. Students will learn how to keep track of multiple pages when printing and render page content correctly.

Lesson

- Printing Features that Are Supported by .NET Framework 2.0.
- Drawing Print Document Content by Using GDI+.

Lab 12: Creating Customized Print Components

- Exercise 1. Printing a Report by Using GDI+.
- Exercise 2. Creating a Customized Print Preview Dialog Box (if time permits).

After completing this unit, students will be able to:

- Describe the printing features that are supported by .NET Framework 2.0.
- Explain how to use GDI+ to draw print document content.
- Print content by using GDI+.
- Describe best practices for creating custom print components.

Unit 13: Performing Drag-and-Drop Operations and Implementing Clipboard Support

This unit introduces the properties, methods, and events that can be used to implement drag-and-drop functionality in a Microsoft Windows Forms application. Students will learn how to start and finish drag-and-drop operations and, specifically, how to implement drag-and-drop operations with a TreeView control. In addition, this unit demonstrates how to use the Clipboard to store and retrieve data.

Lessons

- Drag-and-Drop Operations in Windows Forms Applications.
- Adding Clipboard Support in Windows Forms Applications.

Lab 13: Performing Drag-and-Drop Operations and Implementing Clipboard Support

- Exercise 1. Implementing Drag-and-Drop Functionality in a Windows Forms Application.
- Exercise 2. Adding Clipboard Support to an Application.
- Exercise 3. Performing Drag-and-Drop Operations by Using TextBox and PictureBox Controls (if time permits).

After completing this unit, students will be able to:

- Describe the phases of a drag-and-drop operation.
- Describe the methods that you can use to provide Clipboard support.
- Implement drag-and-drop functionality in a Windows Forms application.
- Add Clipboard support to an application.

Unit 14: Performing Asynchronous Tasks by Using Multithreaded Techniques

This unit demonstrates how to create Microsoft Windows Forms applications that can run tasks in the background. It explains how to make use of the asynchronous methods and other features of components that support the Asynchronous Pattern for Components. Students will also learn how to use the classes in the System.Threading namespace to run one or more tasks in the background by using multiple threads in an application.

Lessons

- Asynchronous Programming in Windows Forms Applications.
- Creating Thread-Safe Applications.

Lab 14: Performing Asynchronous Tasks by Using Multithreaded Techniques

- Exercise 1. Loading a Bitmap Asynchronously.
- Exercise 2. Performing Calculations by Using Multithreading.

After completing this unit, students will be able to:

- Explain the concepts and processes that are involved in performing asynchronous programming.
- Load a bitmap into a Windows Forms application asynchronously.
- Perform multiple simultaneous calculations on a form by using multithreading.

Unit 15: Enhancing the Presentation of Windows Forms Applications

This unit describes several of the features that can be used when creating professional-looking

applications. Students will learn how to build a Windows Form that has the appearance of Microsoft Office Outlook and how to configure a customized master/detail DataGridView control. In addition, this unit explains how to incorporate the PropertyGrid component and application settings features that enable users to edit and save their preferences.

Lessons

- Enhancing Application User Interfaces.
- Customizing the DataGridView Control.
- Application Settings and the PropertyGrid Control.

Lab 15: Enhancing the Presentation of Windows Forms Applications

- Exercise 1. Programming the DataGridView Control.
- Exercise 2. Viewing and Persisting Application Settings by Using the PropertyGrid Control.
- Exercise 3. Implementing a User Interface in the Style of Outlook by Using RAD Features.

After completing this unit, students will be able to:

- Describe several key features involved in enhancing an application user interface.
- Describe how to customize the DataGridView control.
- Explain the concepts of Windows Forms application settings and the PropertyGrid control.
- Program the DataGridView control.
- View and persist application settings by using the PropertyGrid control.
- Implement a user interface in the style of Outlook by using rapid application development (RAD) features.

To register or for more information call our office **(208) 898-9036** or email register@leapfoxlearning.com



Microsoft Partner
Silver Learning