

# Advanced Windows Forms Technologies with Microsoft Visual Studio 2005

Workshop 2547A: Two days; Instructor-Led

## Introduction

This two-day instructor-led workshop provides students with the knowledge and skills to develop advanced 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 corporate or independent software vendor (ISV) application developers who have a desire to learn more about specific technology areas in Windows application development.

## Objectives

After completing this workshop, students will be able to:

- 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:

- Have attended or studied Workshop 2546A, Core Windows Forms Technologies with Visual Studio 2005 or possess equivalent knowledge and skills.
- Be able to manage a solution environment using the Visual Studio 2005 integrated development environment (IDE) and tools.
- Be able to program an application using a .NET Framework 2.0 compliant language, including the use of delegates and events.
- Understand advanced concepts including serialization, reflection, application domains, and multithreading.

## Workshop Outline

### Unit 1: 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.

## **Unit 2: 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.

## **Unit 3: 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.

## **Unit 4: 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.

## **Unit 5: 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.

## **Unit 6: 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.