

Windows® Presentation Foundation (WPF) and Silverlight Introduction: Hands-On - 4 Days

Course 975 Overview

- You Will Learn How To**
- Create compelling, modern user experiences with Windows Presentation Foundation (WPF) and Silverlight
 - Employ Visual Studio, XAML, C# and Visual Basic to create WPF and Silverlight applications
 - Deploy cross-platform .NET applications for Windows, Mac OS X and other systems using Silverlight
 - Manage complex screen layouts that support varying devices, screen sizes and resolutions
 - Design visually appealing, dynamic user interfaces with styles, animations, transformations and triggers
 - Simplify complex architectures with data-binding and the Model-View-ViewModel design pattern (MVVM)

Course Benefits Windows Presentation Foundation (WPF) is the next generation platform for creating user interfaces (UIs) with the Microsoft .NET Framework. Using WPF, developers and designers can easily create programs that are dynamic, data-driven and visually appealing. In this course, you program applications that integrate the major components of WPF. You also learn how to deploy your .NET applications to multiple platforms with Silverlight.

Who Should Attend .NET and other developers who want to leverage WPF and Silverlight to build modern, visual applications for Windows, Mac OS X and other environments. .NET programming experience at the level of Course 502, "Programming with .NET Introduction," is assumed.

Hands-On Training You gain hands-on experience building WPF and Silverlight applications with XAML, VB and C#. Exercises include:

- Creating WPF and Silverlight projects using Visual Studio
- Laying out XAML user interfaces with WPF and Silverlight containers and controls
- Enhancing visual appeal with styles, templates, triggers, animations and transformations
- Synchronizing user interfaces to bindable data sources
- Refactoring application architecture to the MVVM design pattern

Windows® Presentation Foundation (WPF) and Silverlight Introduction: Hands-On - 4 Days

Course 975 Outline

Programming WPF with XAML and Visual Studio

Creating WPF applications in Visual Studio

- Advantages of WPF and Silverlight
- Setting application and window properties
- Implementing behaviors in code-behind files

Coding in XAML

- Adding controls to containers
- Setting properties using attribute and element syntax
- Referencing external namespaces using XAML
- Handling bubbling and tunneling routed events

Architecting WPF Programs

Building browser-based and navigation applications

- Loading XAML pages into the browser
- Moving between pages with the navigation service
- Hyperlinking to other pages and websites
- Passing data between pages
- Navigating page history

Deploying WPF applications with ClickOnce

- Publishing to the web
- Configuring prerequisites and updates
- Securing applications deployed with ClickOnce

Crafting Cross-Platform Silverlight UIs

Creating Silverlight web applications

- Coding Silverlight with C#, VB and XAML
- Navigating a Silverlight application
- Employing Silverlight controls
- Exploiting WCF services

Running Silverlight applications outside the browser

- Installing Silverlight applications on Windows and OS X
- Downloading updates
- Detecting the network

Integrating Silverlight with ASP.NET

- Accessing the browser
- Controlling HTML page content
- Invoking JavaScript functions from Silverlight and vice versa

Designing WPF and Silverlight UIs Leveraging WPF controls

- Button
- TextBox
- TextBlock
- Label
- CheckBox
- Slider
- Image
- Menu
- Status bars
- RichTextBox
- Positioning controls in the Grid and Canvas containers
- Silverlight control toolkit

Managing sophisticated layouts

- Defining grid rows and columns
- Assigning controls to grid cells
- Spanning multiple cells
- Resizing rows and columns with the GridSplitter
- Positioning elements with StackPanel and WrapPanel
- Docking controls to windows using the DockPanel

Styling WPF and Silverlight Applications

Altering the appearance of UI elements

- Margin
- Padding
- Fonts
- Background
- Foreground
- Opacity
- Transparency
- Specifying styles using elements

Maintaining consistent styles with resources

- Naming and extending control styles
- Packaging styles in resource dictionaries
- Customizing controls with templates

Adding images and graphics to the UI

- Drawing shapes and gradients
- Converting scalable vector graphics to XAML
- Programming user controls and dependency properties

Automating Data Access with Data Binding

Binding control properties with XAML

- ItemsControl
- ComboBox
- ListBox
- Formatting content with data templates
- Enhancing forms with scrollable and expandable regions

Accessing data using data providers

- Displaying XML data with the XMLDataProvider and XPath
- Accessing collections using the ObjectDataProvider
- Programming custom type-converters

Model-View-ViewModel (MVVM) design pattern

- Binding to the ViewModel class
- Validating user input
- Implementing the INotifyPropertyChanged interface
- Refreshing records automatically using ObservableCollection

Streamlining event handling with commands and attached behaviors

- Connecting elements to built-in WPF commands
- Programming custom-routed commands
- Leveraging relay and delegate commands
- Adding attached properties and behaviors