70-502

TS: Microsoft .NET Framework 3.5 - Windows Presentation Foundation Application Development

Q&A

DEMO Version

Important Note
Please Read Carefully

For demonstration purpose only, this free version Chinatag study guide contains **10** full length questions selected from our full version products which have more than **200** questions each.

This Study guide has been carefully written and compiled by Chinatag certification experts. It is designed to help you learn the concepts behind the questions rather than be a strict memorization tool. Repeated readings will increase your comprehension.

For promotion purposes, all PDF files are **not** encrypted. Feel free to distribute copies among your friends and let them know Chinatag website.

Study Tips

This product will provide you questions and answers along with detailed explanations carefully compiled and written by our experts. Try to understand the concepts behind the questions instead of cramming the questions. Go through the entire document at least twice so that you make sure that you are not missing anything.

Latest Version

We are constantly reviewing our products. New material is added and old material is revised. Free updates are available for 90 days after the purchase. You should check the products page on the http://www.chinatag.com website for an update 3-4 days before the scheduled exam date.

Please tell us what you think of our products. We appreciate both positive and critical comments as your feedback helps us improve future versions. Feedback on specific questions should be send to feedback@chinatag.com.

Thanks for purchasing our products and look forward to supplying you with all your Certification training needs.

Good studying!

Technical and Support Team
Chinatag LLC.

Section 70-502 (VB)

Question: 1
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. You create a window for the application. You need to ensure that the following requirements are met: An array of strings is displayed by using a ListBox control in a two-column format. The data in the ListBox control flows from left to right and from top to bottom. What should you do?

A. Use a ListBox control defined in the following manner. <ListBox Name="myList"> <ListBox.ItemsPanel> <ItemsPanelTemplate> <UniformGrid Columns="2"/> </ItemsPanelTemplate> </ListBox.ItemsPanel></ListBox> Use the following VB.net code to associate the array of strings to the ListBox control. myList.ItemsSource = arrayOfString
B. Use a ListBox control defined in the following manner. <ListBox Name="myList"> <ListBox.ItemsPanel> <ItemsPanelTemplate> <StackPanel /> </ItemsPanelTemplate> </ListBox.ItemsPanel></ListBox> Use the following vb.net code to associate the array of strings to the ListBox control. myList.ItemsSource = arrayOfString
C. Use a ListBox control defined in the following manner. <ListBox Name="myList"> <ListBox.ItemsPanel> <ItemsPanelTemplate> <WrapPanel /> </ItemsPanelTemplate> </ListBox.ItemsPanel></ListBox> Use the following vb.net code to associate the array of strings to the ListBox control. myListView.ItemsSource = arrayOfString
D. Use a ListBox control defined in the following manner. <ListBox Name="myList"> <ListBox.ItemsPanel> <ItemsPanelTemplate> <Grid> <Grid.ColumnDefinitions> <ColumnDefinition /> <ColumnDefinition /> </Grid.ColumnDefinitions> </Grid> </ItemsPanelTemplate> </ListBox.ItemsPanel></ListBox> Use the following vb.net code to associate the array of strings to the ListBox control. myList.ItemsSource = arrayOfString

Answer: A

Question: 2
You create a form by using Windows Presentation Foundation and Microsoft .NET Framework 3.5. The form contains a status bar. You plan to add a ProgressBar control to the status bar. You need to ensure that the ProgressBar control displays the progress of a task for which you cannot predict the completion time. Which code segment should you use?

A. progbar.IsIndeterminate = True
B. progbar.IsIndeterminate = False
C. progbar.HasAnimatedProperties = True
D. progbar.HasAnimatedProperties = False

Answer: A

Question: 3
You are converting a Windows Forms application to a Windows Presentation Foundation (WPF) application. You use Microsoft .NET Framework 3.5 to create the WPF application. The WPF application will reuse 30 forms of the Windows Forms application. The WPF application contains the following class definition.

```
Public Class OwnerWindow
Implements System.Windows.Forms.IWin32Window
Private handle_Renamed As IntPtr
Public Property Handle() As IntPtr _
Implements System.Windows.Forms.IWin32Window.Handle
Get
Return handle_Renamed
```

```
End Get
Set(ByVal value As IntPtr)
handle_Renamed = value
End Set
End Property
End Class
```

You write the following code segment in the WPF application. (Line numbers are included for reference only.)

```
01 Public Function LaunchWindowsFormsDialog(ByVal dialog As _
02 System.Windows.Forms.Form, ByVal wpfParent As Window) As _
03 System.Windows.Forms.DialogResult
04 Dim helper As New
05 System.Windows.Interop.WindowInteropHelper(wpfParent)
07 Dim owner As New OwnerWindow()
08
09 End Function
```

You need to ensure that the application can launch the reusable forms as modal dialogs.
Which code segment should you insert at line 08?

A. owner.Handle = helper.OwnerDim db As New System.Windows.Forms.DialogResult()Return db
B. owner.Handle = helper.OwnerReturn dialog.ShowDialog(owner)
C. owner.Handle = helper.OwnerDim result As Nullable(Of Boolean) = wpfParent.ShowDialog()If result.HasValue Then Return If(result.Value, System.Windows.Forms.DialogResult.OK, _ System.Windows.Forms.DialogResult.Cancel)Else Return System.Windows.Forms.DialogResult.CancelEnd If
D. owner.Handle = helper.HandleDim result As Nullable(Of Boolean) = wpfParent.ShowDialog()If result.HasValue Then Return If(result.Value, System.Windows.Forms.DialogResult.OK, _ System.Windows.Forms.DialogResult.Cancel)Else Return System.Windows.Forms.DialogResult.CancelEnd If

Answer: B

Question: 4
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. You create a window in the application. You plan to select a layout control to host the elements that you add to the window.

You need to select a control that meets the following requirements with its default properties:
The elements stretch horizontally to occupy the available width of the window.
The elements do not stretch vertically.

Which control should you use?

A. The Grid control
B. The Canvas control
C. The WrapPanel control
D. The StackPanel control

Answer: D

Question: 5
You create a form by using Windows Presentation Foundation. You use Microsoft .NET Framework 3.5 to create the form. You add a ContextMenu control to the text box named myText on the form. You add the following menu items to the control:

Copy
Paste

You need to ensure that the following requirements are met:

You can copy and paste text.
The ContextMenu items have input text gestures.
You can copy and paste text by either clicking the menu items or by using keyboard shortcuts.
You want to achieve this goal by using the least possible code.

Which code fragment should you use?

A. <TextBox Name="myText"> <TextBox.ContextMenu> <ContextMenu> <MenuItem Command="ApplicationCommands.Copy"/> <MenuItem Command="ApplicationCommands.Paste"/> </ContextMenu> </TextBox.ContextMenu></TextBox>
B. <TextBox Name="myText"> <TextBox.CommandBindings> <CommandBinding Command="ApplicationCommands.Copy"/> <CommandBinding Command="ApplicationCommands.Paste"/> </TextBox.CommandBindings> <TextBox.ContextMenu> <ContextMenu> <MenuItem Command="ApplicationCommands.Copy" Header="Copy" InputGestureText="Ctrl+C"/> <MenuItem Command="ApplicationCommands.Paste" Header="Paste" InputGestureText="Ctrl+V"/> </ContextMenu> </TextBox.ContextMenu></TextBox>
C. <TextBox Name="myText"> <TextBox.InputBindings> <KeyBinding Command="ApplicationCommands.Copy" Modifiers="Control" Key="C"/> <KeyBinding Command="ApplicationCommands.Paste" Modifiers="Control" Key="V"/> </TextBox.InputBindings> <TextBox.ContextMenu> <ContextMenu> <MenuItem Command="ApplicationCommands.Copy" Header="Copy" InputGestureText="Ctrl+C"/> <MenuItem Command="ApplicationCommands.Paste" Header="Paste" InputGestureText="Ctrl+V"/> </ContextMenu> </TextBox.ContextMenu></TextBox>
D. <TextBox Name="myText"> <TextBox.InputBindings> <KeyBinding Command="ApplicationCommands.Copy" Modifiers="Control" Key="C"/> <KeyBinding Command="ApplicationCommands.Paste" Modifiers="Control" Key="V"/> </TextBox.InputBindings> <TextBox.CommandBindings> <CommandBinding Command="ApplicationCommands.Copy"/> <CommandBinding Command="ApplicationCommands.Paste"/> </TextBox.CommandBindings> <TextBox.ContextMenu> <ContextMenu> <MenuItem Command="ApplicationCommands.Copy" Header="Copy"/> <MenuItem Command="ApplicationCommands.Paste" Header="Paste"/> </ContextMenu> </TextBox.ContextMenu></TextBox>

Answer: A

Question: 6

You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. You need to perform the following tasks:

Add a control to a window by using the following XAML code fragment.
<local:RedTextControl Background="Yellow" />
Ensure that the background color of the control is yellow.

What should you do?

A. Add the following code segment to a code-behind file. Public Class RedTextControl Inherits ContentControl Public Sub New() Dim tb As New TextBlock() tb.Foreground = Brushes.Red tb.Text = "Hello" Me.AddChild(tb) End SubEnd Class
B. Add the following code segment to a code-behind file. Public Class RedTextControl Inherits Control Public Sub New() Dim tb As New TextBlock() tb.Foreground = Brushes.Red tb.Text = "Hello" Me.AddLogicalChild(tb) End SubEnd Class
C. Add the following XAML code fragment to an XAML file. <UserControl x:Class="RedTextControl" xmlns="..." xmlns:x="..."> <DockPanel> <TextBlock Foreground="Red" Text="Hello" /> </DockPanel></UserControl> Add the following code segment to a code-behind file. Partial Friend Class RedTextControl Public Sub New() InitializeComponent();
Background.CoerceValue(TextBlock.BackgroundProperty); End SubEnd Class
D. Add the following XAML code fragment to an XAML file. <UserControl x:Class="RedTextControl"
xmlns="..." xmlns:x="..."> <DockPanel> <TextBlock Foreground="Red" Text="Hello" /> </DockPanel></UserControl> Add the following code segment to a code-behind file. Partial Friend Class RedTextControl Inherits UserControl Public Sub New() InitializeComponent() End SubEnd Class

Answer: D

Question: 7
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. You plan to add a Button control to a Canvas control. You need to ensure that exactly 10 device-independent pixels are present between the right side of the Button control and the right side of the Canvas control. Which XAML code fragment should you use?

A. <Canvas Margin="10"> <Button>I'm a button</Button></Canvas>
B. <Canvas> <Button Canvas.Left="10" Canvas.Right="10"> I'm a button </Button></Canvas>
C. <Canvas Margin="10"> <Button HorizontalAlignment="Right"> I'm a button </Button></Canvas>
D. <Canvas> <Button Canvas.Top="10" Canvas.Right="10"> I'm a button </Button></Canvas>

Answer: D
Question: 8
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
You create a dialog window composed of a TextBox control and a Button control by using the following XAML code fragment.

<TextBox Width="200" AcceptsReturn="True" />
<Button Width="80" Click="Button_Click">OK</Button>

You need to ensure that each time the user presses the ENTER key, the click event of the OK button is raised.

Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

A. Add the IsDefault="True" attribute to the Button element.
B. Add the IsTabStop="True" attribute to the Button element.
C. Add the IsHitTestVisible="True" attribute to the Button element.
D. Add the Focusable="False" attribute to the TextBox element.
E. Add the AcceptsTab="True" attribute to the TextBox element.
F. Set the AcceptsReturn attribute value of the TextBox element to False.

Answer: A, F

Question: 9
You are creating a Windows Presentation Foundation (WPF) application by using Microsoft .NET Framework 3.5. You plan to implement a search functionality for a text editor. You write the following code segment for the WPF screen.

```
<DockPanel>
<WrapPanel DockPanel.Dock="Top">
<TextBox Name="tbxTextToFind" Width="200" />
<Button Name="btnFind" Width="80"
Click="btnFind_Click">Find</Button>
</WrapPanel>
<RichTextBox Name="rtbText" />
</DockPanel>
```

You need to ensure that on clicking the Find button, the value of the tbxTextToFind text box is selected in the rtbText control.

Which code segment should you use?

A. Dim doc As FlowDocument = rtbText.DocumentDim text As String = (New TextRange(doc.ContentStart, _doc.ContentEnd)).TextDim index As Integer = text.IndexOf(tbxTextToFind.Text)Dim start As TextPointer = doc.ContentStart.GetPositionAtOffset(index)Dim [end] As TextPointer = _start.GetPositionAtOffset(Len(tbxTextToFind.Text))rtbText.Selection.Select(start, [end])
B. Dim cur As TextPointer = rtbText.Document.ContentStartWhile Not cur Is NothingDim [end] As TextPointer = _cur.GetPositionAtOffset(Len(tbxTextToFind.Text))If Not [end] Is Nothing ThenDim search As New TextRange(cur, [end])If search.Text = tbxTextToFind.Text ThenrtbText.Selection.Select(search.Start, search.End)Exit WhileEnd IfEnd Ifcur = cur.GetNextContextPosition(LogicalDirection.Forward)End While
C. Dim cur As TextPointer = rtbText.Document.ContentStartWhile Not cur Is NothingDim [end] As TextPointer = _cur.GetPositionAtOffset(Len(tbxTextToFind.Text))If Not [end] Is Nothing ThenDim search As New TextRange(cur, [end])If search.Text = tbxTextToFind.Text ThenrtbText.Selection.Select(search.Start, search.End)Exit WhileEnd IfEnd Ifcur = cur.GetNextInsertionPosition(LogicalDirection.Forward)End While
D. Dim doc As FlowDocument = rtbText.DocumentDim text As String = (New TextRange(doc.ContentStart,

_doc.ContentEnd)).TextDim index As Integer = text.LastIndexOf(tbxTextToFind.Text)Dim start As TextPointer = _doc.ContentStart.GetPositionAtOffset(index + 1)Dim [end] As TextPointer = _start.GetPositionAtOffset(Len(tbxTextToFind.Text))rtbText.Selection.Select(start, [end])

Answer: C

Question: 10
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. You plan to add a Button control and a StatusBar control to a window.
You need to ensure that the two controls meet the following requirements:

The StatusBar control sticks to the lower edge of the window.
The Button control has the minimum required size to display its content.

Which XAML code fragment should you use?

A. <DockPanel> <StatusBar DockPanel.Dock="Bottom">Information</StatusBar>
<Button>OK</Button></DockPanel>
B. <StackPanel> <Button>OK</Button> <StatusBar>Information</StatusBar></StackPanel>
C. <WrapPanel Orientation="Vertical"> <Button>OK</Button>
<StatusBar>Information</StatusBar></WrapPanel>
D. <DockPanel LastChildFill="False" > <WrapPanel DockPanel.Dock="Top" >
<Button>OK</Button> </WrapPanel> <StatusBar
DockPanel.Dock="Bottom">Information</StatusBar></DockPanel>

Answer: D