Vendor: Microsoft

Exam Code: 70-502

Exam Name: TS: Microsoft .NET Framework 3.5, Windows Presentation Foundation Application Development

Version: DEMO

QUESTION 1
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
The application has a TreeView class that builds the directory tree for a given directory.
You write the following class.

```
class Folder
{
public string Name
{
get{...}
}
public List<Folder> Subfolders
{
get{}
}
}
```

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

```
01 TreeView tree=new TreeView();
02 Folder folder=new Folder(@"C:\");
04 FrameworkElementFactory labelFactory=new
05 FrameworkElementFactory(typeof(TextBlock));
07 template.VisualTree=labelFactory;
08 tree.ItemTemplate=template;
09 tree.ItemsSource=folder.Subfolders;
```

You need to ensure that the TreeView class displays nodes that correspond to the child folders of the C:\ drive.
Which code segments should you insert at lines 03 and 06?

A. Insert the following code segment at line 03.
HierarchicalDataTemplate template=new
HierarchicalDataTemplate(folder);
template.ItemsSource = new Binding("Subfolders");
Insert the following code segment at line 06.
labelFactory.SetBinding(TextBlock.TextProperty, new Binding("Name"));

B. Insert the following code segment at line 03.
HierarchicalDataTemplate template = new
HierarchicalDataTemplate(typeof(Folder));
template.ItemsSource = new Binding("Subfolders");
Insert the following code segment at line 06.
labelFactory.SetBinding(TextBlock.TextProperty, new Binding("Name"));

C. Insert the following code segment at line 03.
HierarchicalDataTemplate template=new
HierarchicalDataTemplate("Folder");
template.ItemsSource = new Binding("Name");
Insert the following code segment at line 06.
labelFactory.SetBinding(TextBlock.TextProperty, new Binding("Subfolders"));

D. Insert the following code segment at line 03.
HierarchicalDataTemplate template = new
HierarchicalDataTemplate("Folder");
template.ItemsSource = new Binding("Folder.Subfolders");
Insert the following code segment at line 06.

labelFactory.SetBinding(TextBlock.TextProperty, new Binding("Folder.Name"));

Answer: B

QUESTION 2
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
You are creating an XAML page for the application. The page will display information about cars. The information is stored in a static resource named cars.
You need to ensure that the page can display the image of each car in a ListBox control.
Which XAML code fragment should you use?

A. <ListBox ItemsSource="{Binding Source={StaticResource cars}}">
<ListBoxItem> <Image Source="{Binding Path=Path}"/> </ListBoxItem></ListBox>
B. <ListBox ItemsSource="{Binding Source={StaticResource cars}}">
<ListBoxItem> <DataTemplate> <Image Source="{Binding Path=Path}"/> </DataTemplate>
</ListBoxItem></ListBox>
C. <ListBox ItemsSource="{Binding Source={StaticResource cars}}">
<ListBox.ItemTemplate> <DataTemplate> <Image Source="{Binding Path=Path}"/>
</DataTemplate> </ListBox.ItemTemplate></ListBox>
D. <ListBox ItemsSource="{Binding Source={StaticResource cars}}">
<ListBox.ItemTemplate> <DataTemplate> <Image Source="{Binding Source={StaticResource cars}, Path=Path}"/> </DataTemplate> </ListBox.ItemTemplate></ListBox>

Answer: C

QUESTION 3
You are creating a Windows Presentation Foundation application to track customer orders.
You use Microsoft .NET Framework 3.5 to create the application. The application contains an Order class. The Order class has two public properties named OrderDate and OrderID. The application also contains an ObservableCollection collection of Order objects.
This collection is named OrderItems.
You write the following XAML code fragment.

```
<Window.Resources>
<CollectionViewSource
Source="{Binding Source={x:Static Application.Current},
Path=OrderItems}"
x:Key="orders" />
</Window.Resources>
```

You add two ListBox controls to select orders based on the OrderID or the OrderDate properties. One ListBox control displays order dates and the other ListBox control displays order numbers.
You need to bind the two ListBox controls so that the item selected in one control is reflected in the other control.
Which XAML code fragment should you use?

A. <ListBox Name="lstOrder" DisplayMemberPath="OrderID" ItemsSource="{Binding Source={StaticResource orders}}"/>
<ListBox Name="lstDate" DisplayMemberPath="OrderDate" ItemsSource="{Binding Source={StaticResource orders}}"/>
B. <ListBox Name="lstOrder" DisplayMemberPath="OrderID" ItemsSource="{Binding Source={StaticResource orders}}"/>

<ListBox Name="lstDate" DisplayMemberPath="OrderDate" ItemsSource="{Binding ElementName=lstOrder, Path=OrderDate}"/>

C. <ListBox Name="lstOrder" IsSynchronizedWithCurrentItem="True" DisplayMemberPath="OrderID" ItemsSource="{Binding Source={StaticResource orders}}"/> <ListBox Name="lstDate" IsSynchronizedWithCurrentItem="True" DisplayMemberPath="OrderDate" ItemsSource="{Binding Source={StaticResource orders}}"/>

D. <ListBox Name="lstOrder" IsSynchronizedWithCurrentItem="True" DisplayMemberPath="OrderID" ItemsSource="{Binding Source={StaticResource orders}}"/> <ListBox Name="lstDate" SynchronizedWithCurrentItem="True" DisplayMemberPath="OrderDate" ItemsSource="{Binding ElementName=lstOrder, Path=OrderDate}"/>

Answer: C

QUESTION 4
You are creating a Windows Presentation Foundation GUI application by using Microsoft .NET Framework 3.5.
You use the following code fragment to create a simple calculator application named MyApp. (Line numbers are included for reference only.)

```
01 <StackPanel Orientation="Horizontal">
02 <StackPanel.Resources>
03 <local:Calc x:Key="calc" xmlns:local="clr-namespace:MyApp"/>
04 </StackPanel.Resources>
05 <TextBox Name="first"/>
06 <TextBox Name="second"/>
07 <TextBlock Name="result">
08 <TextBlock.Text>
10 </TextBlock.Text>
11 </TextBlock>
12 </StackPanel>
```

You need to ensure that the TextBlock control named result displays the sum of the values contained in the TextBox controls named first and second.
Which two tasks should you perform? (Each correct answer presents part of the solution. Choose two.)

A. Insert the following code fragment at line 09.
<Binding Source="first" RelativeSource="second" Path="Text" Converter="{StaticResource calc}">

B. Insert the following code fragment at line 09.
<MultiBinding Converter="{StaticResource calc}">
<Binding ElementName="first" Path="Text"/> <Binding ElementName="second" Path="Text"/></MultiBinding>

C. Insert the following code fragment at line 09.
<MultiBinding Converter="{StaticResource calc}">
<Binding Source="first" Path="Text"/>
<Binding Source="second" Path="Text"/></MultiBinding>

D. Insert the following code fragment at line 09.
<MultiBinding> <Binding Source="first" Path="Text" Converter="{StaticResource calc}"/>
<Binding Source="second" Path="Text" Converter="{StaticResource calc}"/></MultiBinding>

E. Create the Calc class that implements the IMultiValueConverter interface.
Include the following code segment in the Convert method. int firstval, secondval;if(int.TryParse(values[0].ToString(), out firstval) &&

int.TryParse(values[1].ToString(),out secondval)){ return (firstval + secondval).ToString();}
return "NaN";

F. Create the Calc class that implements the IMultiValueConverter interface.
Include the following code segment in the the Convert method. int firstval, secondval;
if (int.TryParse(first.Text.ToString(), out firstval) && int.TryParse(second.Text.ToString(), out secondval)){ return (first + second).ToString();}
return "NaN";

Answer: BE

QUESTION 5
You are creating a Windows Presentation Foundation application for a car dealer.
You use Microsoft .NET Framework 3.5 to create the application.
You are creating a window that will display a list of available cars.
The list will be sorted on the basis of the Make property.
You try to run the XAML code fragment for the window. The following section of the code fragment does not compile. (Line numbers are included for reference only.)

```
01 <Window.Resources>
02 ...
03 <CollectionViewSource x:Key="vw"
04 Source="{StaticResource cars}">
05 <CollectionViewSource.SortDescriptions>
06 <SortDescription PropertyName="Make" />
07 </CollectionViewSource.SortDescriptions>
08 </Window.Resources>
```

You receive the following error message:

```
"Type 'SortDescriptions' was not found."
```

You need to ensure that the XAML code fragment sorts the list of available cars.
What should you do?

A. Replace line 06 of the XAML code fragment with the following code fragment.
<col:SortDescription PropertyName="Make"
xmlns:col="clr-namespace:System.CodeDom;assembly=System"/>
B. Replace line 06 of the XAML code fragment with the following code fragment.
<col:SortDescription PropertyName="Make"
xmlns:col="clr-namespace:System.ComponentModel;assembly=WindowsBase"/>
C. Add the following code segment immediately after the IntializeComponent method call in the constructor. BindingListCollectionView view = CollectionViewSource.
GetDefaultView(lst.ItemsSource) as BindingListCollectionView;if (view != null)
view.SortDescriptions.Add(new SortDescription("Make", ListSortDirection.Ascending));
D. Remove lines 03 through 07 from the XAML code fragment. Add the following code segment immediately after the InitializeComponent method call in the constructor.
BindingListCollectionView view = CollectionViewSource. GetDefaultView(lst.ItemsSource) as BindingListCollectionView;if (view != null) view.SortDescriptions.Add(new SortDescription("Make", ListSortDirection.Ascending));

Answer: B

QUESTION 6

You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
The XAML page for the application will display employee information from an XML file.
The following code fragment is an excerpt from the XML file.

```
<Employee>
<Manager FirstName="Adam" LastName="Barr" Photo="E1.jpg"/> <Engineer
FirstName="Mu" LastName="Han" Photo="E2.jpg"/> ...
</Employee>
```

The file is exposed as a resource named employees.
You add the following templates under the Window.Resources element.

```
<DataTemplate x:Key="Manager">
<Image Source="{Binding XPath=@Photo}" Height="50"/> </DataTemplate>
<DataTemplate x:Key="Engineer">
<Image Source="{Binding XPath=@Photo}" Height="40"/> </DataTemplate>
```

You need to ensure that the selection of templates is dependent on the XML element being rendered.
Which two tasks should you perform? (Each correct answer presents part of the solution. Choose two.)

A. Add the following ListBox control to the XAML code fragment for the window.
<ListBox ItemsSource="{Binding Source={StaticResource employees}, XPath=/Employee/*}"/>
B. Add the following template to the Windows.Resources element.
<HierarchicalDataTemplate
x:Key="Employee" ItemsSource="{Binding XPath=/Employee/*}">
<TextBlock Text="Employees" FontSize="20" /></HierarchicalDataTemplate>
C. Add the following template to the Windows.Resources element.
<HierarchicalDataTemplate
x:Key="Employee" ItemsSource="{Binding XPath=*}"> <TextBlock Text="Employees" FontSize="20" /></HierarchicalDataTemplate>
D. Add the following ListBox control to the XAML code fragment for the page. <ListBox ItemsSource="{Binding Source={StaticResource employees}, XPath=/Employee/*}">
<ListBox.ItemTemplateSelector> <local:EmployeeTemplateSelector xmlns:local="clr-namespace:AppNamespace"/>
</ListBox.ItemTemplateSelector></ListBox>
E. Add the following class to the code-behind file. public class EmployeeTemplateSelector: DataTemplateSelector { public override DataTemplate SelectTemplate(object item, DependencyObject container){ XmlElement data = item as XmlElement; return ((FrameworkElement)container).
FindResource(data.LocalName) as DataTemplate; }}

Answer: DE

QUESTION 7
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
You create the following classes.

```
public class Car {
public string Model;
public string ImagePath;
```

```
}
public class Cars : Collection<Car> {
protected override void ClearItems()...
protected override void InsertItem(int index, Car item)...
}
```

You bind a Label control and a ListBox control to the Cars class by using the following XAML code fragment.

```
<Label Content="{Binding Source={StaticResource cars},
Path=Count}"/> ...
<Image Source="{Binding Source={StaticResource cars},
Path=ImagePath}"> ...
```

You need to ensure that the Cars class can propagate changes to the bound controls.
How should you modify the Cars class?

A. Implement the IDataObject interface.
B. Replace the Collection<Car> inheritance with the ObservableCollection<Car> inheritance.
C. Add a public Count property that returns an integer value and add a public ImagePath property that returns a string value.
D. Implement the INotifyPropertyChanged interface and add the following code segment. public event PropertyChangedEventHandler PropertyChanged;protected void OnPropertyChanged(string propertyName) { PropertyChangedEventHandler handler = PropertyChanged;
if (this.PropertyChanged != null){ PropertyChanged (this, new PropertyChangedEventArgs(propertyName)); }} Call the OnPropertyChanged("Count") and the OnPropertyChanged("ImagePath") methods from the ClearItems and the InsertItem methods respectively.

Answer: B

QUESTION 8
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
The application processes hospital medical records. The application contains a class named Patient that has a property named Age.
You enter information about the patient in a window named Medical Form.
In the window, you enter the age of the patient in a textbox named txtAge.
You need to ensure that the application meets the following requirements:

```
- It updates the Age property automatically when you enter the age of
the patient in the txtAge textbox.
- It updates the txtAge textbox automatically when you change the Age
property of the Patient class.
```

Which XAML code fragment should you use?

A. <TextBox Name="txtAge"> <TextBox.Resources> <src:Patient x:Key="AgeResource" /> </TextBox.Resources> <TextBox.Text> <Binding Source = "{StaticResource AgeResource}" Mode="OneWayToSource" Path="Age" /> </TextBox.Text></TextBox>
B. <TextBox Name="txtAge"> <TextBox.Resources> <src:Patient x:Key="AgeResource" /> </TextBox.Resources> <TextBox.Text> <Binding Source = "{StaticResource AgeResource}" Mode="OneWay" Path="Age" /> </TextBox.Text></TextBox>

C. <TextBox Name="txtAge"> <TextBox.Resources> <src:Patient x:Key="AgeResource" /> </TextBox.Resources> <TextBox.Text> <Binding Source = "{StaticResource AgeResource}" Mode="Default" Path="Age" /> </TextBox.Text></TextBox>
D. <TextBox Name="txtAge"> <TextBox.Resources> <src:Patient x:Key="AgeResource" /> </TextBox.Resources> <TextBox.Text> <Binding Source = "{StaticResource AgeResource}" Mode="OneTime" Path="Age" /> </TextBox.Text></TextBox>

Answer: C

QUESTION 9
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5. The application contains a Person object with a FirstName property.
You bind the Person object to a data source that is used in several controls.
You need to ensure that the data source is updated whenever the FirstName property changes.
What should you do?

A. Use OneWay data binding.
B. Use OneTime data binding.
C. Create a PropertyChanged event handler in the application.
D. Implement the INotifyPropertyChanged interface in the Person object.

Answer: D

QUESTION 10
You are creating a Windows Presentation Foundation application by using Microsoft .NET Framework 3.5.
The application contains a TextBox control that is data bound to a data source.
You need to ensure that the following requirements are met:

```
- When the data source changes, the TextBox control automatically
reflects the change.
- When the TextBox control changes, the data source is not
automatically updated.
```

What should you do?

A. Use the OneWay binding mode.
B. Use the OneWayToSource binding mode.
C. Use the OneTime binding mode.
D. Use the LostFocus update source trigger.
E. Use the PropertyChanged update source trigger.

Answer: A

Thank You for Trying Our Product

PassLeader Certification Exam Features:

★ More than 99,900 Satisfied Customers Worldwide.

★ Average 99.9% Success Rate.

★ Free Update to match latest and real exam scenarios.

★ Instant Download Access! No Setup required.

★ Questions & Answers are downloadable in PDF format and VCE test engine format.

★ Multi-Platform capabilities - Windows, Laptop, Mac, Android, iPhone, iPod, iPad.

★ 100% Guaranteed Success or 100% Money Back Guarantee.

★ Fast, helpful support 24x7.

View list of all certification exams: http://www.passleader.com/all-products.html

10% Discount Coupon Code: **STNAR2014**