

## Unabridged Pass4sure Microsoft 70-511 PDF Dumps Free Download (181-190)

QUESTION 181 You are developing a Windows Presentation Foundation (WPF) application that displays opportunities from List (Of T) named Leads, where T is a class named Lead. The Lead class contains the properties Title and Revenue. You add a DataGrid control named dgQualifiedLeads to the MainWindow.xaml file. You set the ItemSource property of dgQualifiedLeads to Leads as follows. (Line numbers are included for reference only.)

```
01 Dim Leads As ICollectionView
02 Public Sub New()
03
04     InitializeComponent()
05     Leads = CollectionViewSource.GetDefaultView(
06         New List(Of Lead) (
07             New Lead() { _
08                 New Lead() With {
09                     .Title = "Title1", .Revenue = 1250.78}, _
10                 New Lead() With {
11                     .Title = "Title2", .Revenue = 500.0}, _
12                 New Lead() With {
13                     .Title = "Title3", .Revenue = 2300.33}, _
14                 New Lead() With {
15                     .Title = "Title4", .Revenue = 2989.09}, _
16                 New Lead() With {
17                     .Title = "Title5", .Revenue = 1500.21} _
18             }
19         )
20     )
21     dgQualifiedLeads.ItemsSource = Leads
22
23 End Sub
```

passleader.com

You need to ensure that CollectionViewSource is used to filter the list to display only Lead objects with revenue of more than \$1,000. What should you do? A. Insert the following code at line 22. Leads.Filter = New Predicate (Of Object)(AddressOf FilterOut) Add the following code segment to the code-behind of the MainWindow.xaml file. Public Function FilterOut(ByVal item As Object) As Boolean Dim lead As Lead = TryCast(item, Lead) Return IIf(lead.Revenue < 1000D, True, False) End Function B. Insert the following code at line 22. Leads.Filter = New Predicate (Of Object)(AddressOf FilterOut) Add the following code segment to the code-behind of the MainWindow.xaml file. Public Function FilterOut(ByVal item As Object) As Boolean Dim lead As Lead TryCast(item, Lead) Return IIf(lead.Revenue > 1000D, False, True) End Function C. Insert the following code at line 22. Leads.Filter = New Predicate(Of Object)( Function(s) DirectCast(s, Lead).Revenue > 1000D) D. Insert the following code at line 22. Leads.SortDescriptions.Add(New SortDescription( "Revenue", ListSortDirection.Ascending)) Answer: C QUESTION 182 You are developing a Windows Presentation Foundation (WPF) application that displays pricing and inventory information. A list box's ItemsSource property has decimal and string types. Decimals represent price and strings represent messages such as "Discontinued" The following markup is defined as follows. (Line numbers are included for reference only.)

```
01 <Window ...
02     xmlns:clr="clr-namespace:System;assembly=System.Windows.Controls"
03     <ListBox ItemsSource="{Binding}">
04         <ListBox.Resources>
05
06             </ListBox.Resources>
07         </ListBox>
08     </Window>
```

passle

You need to ensure that data templates are used to format the strings without changes and the decimals as currency. Which markup segment should you insert at line 05? A. <DataTemplate x:Key="clr:String"> <TextBlock Text="{Binding StringFormat=Item Error: {0}}"/> </DataTemplate> <DataTemplate x:Key="clr:Decimal"> <TextBlock Text="{Binding StringFormat=Item Price: {0:C}}"/> </DataTemplate> B. <DataTemplate x:Key="String" Template="clr:String"> <TextBlock Text="{Binding StringFormat=Item Error: {0}}"/> </DataTemplate> <DataTemplate x:Key="Decimal" Template="clr:Decimal"> <TextBlock Text="{Binding StringFormat=Item Price: {0:C}}"/> </DataTemplate> C. <DataTemplate DataType="{x:Type clr:String}"/> <TextBlock Text="{Binding StringFormat=Item Error: {0}}"/> </DataTemplate> <DataTemplate DataType="{x:Type clr:Decimal}"/> <TextBlock Text="{Binding StringFormat=Item Price: {0:C}}"/> </DataTemplate> D. <DataTemplate DataType=,

clr : String"> <TextBlock Text="{Binding StringFormat=Item Error: {0}}"/> </DataTemplate> <DataTemplate  
 DataType="clr: Decimal"> <TextBlock Text="{Binding StringFormat=Item Price: {0: C}}"/> </DataTemplate> Answer: B  
 QUESTION 183 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to create a Windows Presentation  
 Foundation (WPF) application. You create a WPF window in the application. You add the following code segment to the  
 application.

```
Public Class ViewModel
    Public Property Data() As ICollectionView
        Get
        End Get
        Set
        End Set
    End Property
End Class

Public Class BusinessObject
    Public Property Name() As String
        Get
        End Get
        Set
        End Set
    End Property
End Class
```

passleader.com

The DataContext property of the window is set to an instance of the ViewModel class. The Data property of the ViewModel instance is initialized with a collection of BusinessObject objects. You add a TextBox control to the window. You need to bind the Text property of the TextBox control to the Name property of the current item of the ICollectionView of the DataContext object. You also need to ensure that when a binding error occurs, the Text property of the TextBox control is set to N/A . Which binding expression should you use? A.&#160;&#160;&#160; {Binding Path=Data/Name, FallbackValue='N/A'} B.&#160;&#160;&#160; {Binding Path=Data.Name, FallbackValue='N/A'} C.&#160;&#160;&#160; {Binding Path=Data/Name, TargetNullValue='N/A'} D.&#160;&#160;&#160; {Binding Path=Data.Name, TargetNullValue='N/A'} Answer: A  
 QUESTION 184 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application. You write the following code segment. (Line numbers are included for reference only.)

```
01 Public Class Contact
02 Private _contactName As String
03
04 Public Property ContactName() As String
05 Get
06 Return _contactName
07 End Get
08 Set
09 _contactName = value
10 End Set
11 End Property
12
13 End Class
```

passleader.com

You add the following code fragment within a WPF window control.

```
<TextBox>
  <TextBox.Text>
    <Binding Path="ContactName" UpdateSourceTrigger="PropertyChanged">
      <Binding.ValidationRules>
        <DataErrorValidationRule />
      </Binding.ValidationRules>
    </Binding>
  </TextBox.Text>
</TextBox>
```

passleader.com

You need to ensure that the Contact class contains a business rule to ensure that the ContactName property is not empty or NULL. You also need to ensure that the TextBox control validates the input data, Which two actions should you perform? (Each correct answer presents part of the solution. Choose two). A.&#160;&#160;&#160; Replace line 01 with the following code segment. Public Class Contact Implements IDataErrorInfo B.&#160;&#160;&#160; Replace line 01 with the following code segment. Public Class Contact Inherits ValidationRule C.&#160;&#160;&#160; Replace line 01 with the following code segment. Public Class Contact Implements INotifyPropertyChanging D.&#160;&#160;&#160; Add the following code segment at line 03. Public Event PropertyChanging As PropertyChangingEventHandler Modify line 08 with the following code segment: Set If Me.PcpropertyChanging <> Nothing Then Proper:tyChanging(Me, New PropertyChangingEventArgs("ContactName")) End If If String.IsNullOrEmpty(value) Then Throw New ApplicationException("Contact name is required") End If contactName = value End Set E.&#160;&#160;&#160; Add the following code segment at line 12. Public Readonly Property [Error] () As String

Get Throw New Exception( string.Empty ) End Get End Property Public Default Readonly Property Item(columnName As String) As String Get If columnName = "ContactName" AndAlso String.IsNullOrEmpty(Me.ContactName) Then Return "Contact name is required" End If Return Nothing End Get End Property Answer: AE QUESTION 185 You are developing a Windows Presentation Foundation (WPF) application. This application will be used to display customer data to customer service representatives. Phone numbers are stored as ten-digit numbers in the database. The markup is as follows. <TextBlock Text="{Binding Path=PhoneNumber, ConvertersStaticResource PhoneFormatConverter}" /> You need to ensure that phone numbers are displayed in the following format: (###) ### - #### Which markup segment should you use? A.&#160;&#160;&#160; Public Function Convert( ByVal value As Object, ByVal targetType As Type, ByVal parameter As Object, ByVal culture As CultureInfo) As Object Return String.Format("{0:(###)###-####}>", CLng(parameter)) End Function B.&#160;&#160;&#160; Public Function Convert( ByVal value As Object, ByVal targetType As Type, ByVal parameter As Object, ByVal culture As CultureInfo) As Object Return String.Format("{0:(###)###-####}", CLng(value)) End Function C.&#160;&#160;&#160; Public Function ConvertBack( ByVal value As Object, ByVal targetType As Type, ByVal parameter As Object, ByVal culture As CultureInfo) As Object Return String.Format("{0:[###]###-####}", CLng(parameter)) End Function D.&#160;&#160;&#160; Public Function ConvertBack( ByVal value As Object, ByVal targetType As Type, ByVal parameter As Object, ByVal culture As CultureInfo) As Object Return String.Format("{0:[###]###-####}>", CLng(value)) End Function Answer: B QUESTION 186 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application. You add a ListBox control to the application. The ListBox control is data-bound to an instance of a custom collection class of the Product objects named ProductList. You need to ensure that changes to ProductList are automatically reflected in the ListBox control. What should you do? A.&#160;&#160;&#160; Implement the INotifyPropertyChanged interface in the Product class. B.&#160;&#160;&#160; Implement the IQueryable<Product> interface in the ProductList class. C.&#160;&#160;&#160; Extend the DependencyObject class in the Product class. D.&#160;&#160;&#160; Extend the ObservableCollection<Product> class in the ProductList class. Answer: D QUESTION 187 You are developing a Windows Presentation Foundation (WPF) application. You need to display HTML content from a Web page on the WPF form. What should you do? A.&#160;&#160;&#160; Add a FlowDocumentReader control to the design surface. Then create a FlowDocument control. B.&#160;&#160;&#160; Add a DocumentViewer control to the design surface. Then create a FixedDocument control. C.&#160;&#160;&#160; Add a WebBrowser control to the design surface. Then use the Navigate method to navigate the URI object. D.&#160;&#160;&#160; Add a ContentControl control to the design surface. Then reference a WebClient object to return an HTML string. Answer: C QUESTION 188 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application. You need to ensure that the application meets the following requirements: - Displays a menu that is specific to the control selected by the user. - Displays the menu next to the control. Which control should you use? A.&#160;&#160;&#160; Menu B.&#160;&#160;&#160; Popup C.&#160;&#160;&#160; ListBox D.&#160;&#160;&#160; ContextMenu Answer: D QUESTION 189 You use Microsoft .NET Framework 4 to create a Windows Forms application. You need to allow the user interface to use the currently configured culture settings in the Control Panel. Which code segment should you use? A.&#160;&#160;&#160; Thread.CurrentThread.CurrentUICulture = Thread.CurrentThread.CurrentCulture B.&#160;&#160;&#160; Thread.CurrentThread.CurrentCulture = Thread.CurrentThread.CurrentUICulture C.&#160;&#160;&#160; Thread.CurrentThread.CurrentUICulture = CultureInfo.InstalledUICulture D.&#160;&#160;&#160; Thread.CurrentThread.CurrentCulture = CultureInfo.InstalledUICulture Answer: A QUESTION 190 You are developing a Windows Presentation Foundation (WPF) application. You are implementing the security features for a function that requires File IO. The callers of this function that are higher in the stack do not have permission to read the C:\temp directory. The function also accesses other resources that require permission. You need to ensure that the function has the proper permissions to read the C:\temp directory, and that all other resources in the function can still be accessed. Which attribute should you include with the function? A.&#160;&#160;&#160; <FileIOPermissionAttribute ( SecurityAction.Demand, Read:="C:\temp")> B.&#160;&#160;&#160; <FileIOPermissionAttribute( SecurityAction.Assert, Read:="C:\temp")> C.&#160;&#160;&#160; <FileIOPermissionAttribute( SecurityAction.PermitOnly, Read:="C:\temp")> D.&#160;&#160;&#160; <FileIOPermissionAttribute( SecurityAction.InheritanceDemand, Read:="C:\temp")> Answer: B

[Microsoft 70-511 PDF Dumps Free Download](#)

