

ValidBraindumps

Over **61842+** Satisfied Customers

About Us

ValidBraindumps

HOME CERTIFICATIONS HOW TO PAY? GUARANTEE FAQ CART (0)

Test4engine

WE

TRY BEFORE YOU BUY

Download a free sample of any of our exam questions and answers

- > 24/7 customer support, Secure shopping site
- > Free One year updates to match real exam scenarios
- > If you failed your exam after buying our products we will refund the full amount back to you.

IGINE

st version

n exams. Besides for the

ear test questions and

to the highest standards of technical

fronter experts and published authors for

Select a vendor... Select an exam... Your email address Free Download

Try Before You Buy

Download a free sample of any of our exam questions and answers

- > 24/7 customer support, Secure shopping site
- > Free One year updates to match real exam scenarios
- > If you failed your exam after buying our products we will refund the full amount back to you.

<http://www.validbraindumps.com>

Free valid test braindumps for IT certification valid exam

Exam : 70-511

Title : TS: Windows Applications
Development with Microsoft .NET
Framework 4

Vendors : Microsoft

Version : DEMO

NO.1 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

You write the following code fragment.

```
<StackPanel>
<StackPanel.Resources>
<Style TargetType="{x:Type Button}">
<EventSetter Event="Click" Handler="ButtonHandler"/>
</Style>
</StackPanel.Resources>
<Button Name="OkButton">Ok</Button>
<Button Name="CancelButton" Click="CancelClicked">Cancel</Button>
</StackPanel>
```

You need to ensure that the ButtonHandler method is not executed when the user clicks the CancelButton button.

Which code segment should you add to the code-behind file?

- A.

```
private void CancelClicked(object sender, RoutedEventArgs e)
{
    Button btn = (Button)sender;
    btn.Command = null;
}
```
- B.

```
private void CancelClicked(object sender, RoutedEventArgs e) {
    Button btn = (Button)sender;
    btn.IsCancel = true;
}
```
- C.

```
private void CancelClicked(object sender, RoutedEventArgs e) {
    e.Handled = true;
}
```
- D.

```
private void CancelClicked(object sender, RoutedEventArgs e) {
    e.Handled = false;
}
```

Answer: C

NO.2 You create a Windows client application by using Windows Presentation Foundation (WPF).

The application contains the following code fragment.

```
<Window.Resources>
<DataTemplate x:Key="detail">
```

```
<!--...-->
</DataTemplate>
</Window.Resources>
<StackPanel>
<ListBox Name="lbDetails">
</ListBox>
<Button Name="btnDetails">Details</Button>
</StackPanel>
```

You need to assign lbDetails to use the detail data template when btnDetails is clicked.

Which code segment should you write for the click event handler for btnDetails?

- A. lbDetails.ItemsPanel.FindName("detail",lbDetails);
- B. var tmpl = (ControlTemplate)FindResource("detail"); lbDetails.Template = tmpl;
- C. var tmpl = (DataTemplate)FindName("detail"); lbDetails.ItemTemplate = tmpl;
- D. var tmpl = (DataTemplate)FindResource("detail"); lbDetails.ItemTemplate=tmpl;

Answer: D

NO.3 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

You write the following code fragment.

```
<StackPanel TextBox.PreviewTextInput="StackPanel_PreviewTextInput">
<TextBox Name="TxtBoxA"/>
<TextBox Name="TxtBoxB"/>
<TextBox Name="TxtBoxC"/>
</StackPanel>
```

You create an event handler named StackPanel_PreviewTextInput. You also have a collection of strings named Keywords.

You need to ensure that TxtBoxA and TxtBoxB do not contain any of the strings in the Keywords collections.

Which code segment should you use?

```
A. private void StackPanel_PreviewTextInput( object sender, TextCompositionEventArgs e)
{ FrameworkElement feSource = sender as FrameworkElement;
if (feSource.Name == "TxtBoxA" || feSource.Name == "TxtBoxB")
{ foreach(string keyword in Keywords)
{
if(e.Text.Contains(keyword)) {
e.Handled = false;
```

```
return;
```

```
}
```

```
}} e.Handled = true;
```

```
}}
```

```
B. private void StackPanel_PreviewTextInput( object sender, TextCompositionEventArgs e) {
```

```
FrameworkElement feSource = e.Source as FrameworkElement;
```

```
f (feSource.Name == "TextBoxA" || feSource.Name == "TextBoxB")
```

```
f (feSource.Name == "TextBoxA" || feSource.Name == "TextBoxB") {
```

```
foreach(string keyword in Keywords)
```

```
{
```

```
if(e.Text.Contains(keyword)) {
```

```
e.Handled = false;
```

```
return;
```

```
}
```

```
} e.Handled = true;
```

```
C. private void StackPanel_PreviewTextInput( object sender, TextCompositionEventArgs e)
```

```
{
```

```
FrameworkElement feSource = sender as FrameworkElement;
```

```
if (feSource.Name == "TextBoxA" || feSource.Name == "TextBoxB")
```

```
{ foreach(string keyword in Keywords)
```

```
{ if(e.Text.Contains(keyword)) {
```

```
e.Handled = true;
```

```
return; }
```

```
} e.Handled = false;
```

```
}}
```

```
D. private void StackPanel_PreviewTextInput( object sender, TextCompositionEventArgs e)
```

```
{ FrameworkElement feSource = e.Source as FrameworkElement;
```

```
if (feSource.Name == "TextBoxA" || feSource.Name == "TextBoxB")
```

```
{
```

```
foreach(string keyword in Keywords)
```

```
{ if(e.Text.Contains(keyword)) {
```

```
e.Handled = true;
```

```
return;
```

```
}} e.Handled = false;
```

```
}
```

```
}
```

Answer: D

NO.4 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

You create a window that contains a Button control and a MenuItem control. Both controls are labeled

"Add sugar."

The Command properties of the Button and MenuItem controls are set to the same RoutedCommand named AddSugarCommand.

You write the following code segment.

```
private void CanAddSugar (object sender, CanExecuteRoutedEventArgs e) { ... }
```

You need to ensure that when the CanAddSugar method sets e.CanExecute to false, the MenuItem and

Button controls are disabled.

What should you do?

A. Create an event handler for the CanExecuteChanged event of the AddSugarCommand command.

Call the CanAddSugar method from within the event handler.

B. Inherit the AddSugarCommand from the RoutedUICommand class instead of the RoutedCommand class.

Call the CanAddSugar method from within the constructor of the AddSugarCommand command.

C. Add a CommandBinding object to the CommandBinding property of the MenuItem control. Set the CanExecute property of the CommandBinding object to the CanAddSugar method.

D. Add a CommandBinding object to the CommandBindings property of the window.

Set the Command property of CommandBinding to the AddSugarCommand command.

Set the CanExecute property of the CommandBinding object to the CanAddSugar method.

Answer: D

NO.5 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

You want to add an audio player that plays .wav or .mp3 files when the user clicks a button.

You plan to store the name of the file to a variable named SoundFilePath.

You need to ensure that when a user clicks the button, the file provided by SoundFilePath plays.

What should you do?

A. Write the following code segment in the button onclick event.

```
System.Media.SoundPlayer player = new System.Media.
```

```
SoundPlayer(SoundFilePath); player.play();
```

B. Write the following code segment in the button onclick event.

```
MediaPlayer player = new MediaPlayer();
```

```
player.Open(new URI(SoundFilePath), UriKind.Relative)); player.play();
```

C. Use the following code segment from the PlaySound() Win32 API function and call the PlaySound

function in the button onclick event.

```
[sysimport(dll="winmm.dll")]
```

```
public static extern long PlaySound(String SoundFilePath, long hModule, long dwFlags);
```

D. Reference the Microsoft.DirectX Dynamic Link Libraries. Use the following code segment in the button

onclick event.

```
Audio song = new Song(SoundFilePath);
```

```
song.CurrentPosition = song.Duration; song.Play();
```

Answer: B

NO.6 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 ICollectionView Data { get; set; }
```

```
}
```

```
public class BusinessObject
```

```
{
```

```
public string Name { get; set; }
```

```
}
```

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

CollectionView 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. { Binding Path=Data/Name, FallbackValue='N/A' }
- B. { Binding Path=Data.Name, FallbackValue='N/A' }
- C. { Binding Path=Data/Name, TargetNullValue='N/A' }
- D. { Binding Path=Data.Name, TargetNullValue='N/A' }

Answer: A

NO.7 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

The application has a window named MainWindow that has a StackPanel control named sp as the root element.

You want to create a Button control that contains a TextBlock control with the "Save" Text property.

You need to create the control dynamically and add the control to sp.

Which code segment should you write in the constructor of the MainWindow class?

- A. Button btn = new Button();
TextBlock text = new TextBlock();
text.Text = "Save";
btn.Content = text;
sp.DataContext = btn;
- B. Button btn = new Button();
TextBlock text = new TextBlock();
text.Text = "Save";
btn.Content = text;
sp.Children.Add(btn);
- C. Button btn = new Button();
TextBlock text = new TextBlock();
text.Text = "Save";
sp.Children.Add(btn);
sp.Children.Add(text);
- D. Button btn = new Button();
TextBlock text = new TextBlock();
text.Text = "Save";
btn.ContentTemplateSelector.SelectTemplate(text, null);
sp.Children.Add(btn);

Answer: B

NO.8 You use Microsoft .NET Framework 4 to create a Windows Forms application. You plan to use a Windows Presentation Foundation (WPF) control of the UserControl1 type hosted in an ElementHost control named elementHost1.

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

```
01public class WPFInWinForms {
02public WPFInWinForms
03{
04InitializeComponent();
05
06}
07private void OnBackColorChange(object sender, String propertyName, object value)
08{
09ElementHost host = sender as ElementHost;
10System.Drawing.Color col = (System.Drawing.Color)value;
11SolidColorBrush brush =
new SolidColorBrush(System.Windows.Medi
a.Color.FromRgb(col.R, col.G, col.B));
12UserControl1 uc1 = host.Child as UserControl1;
13uc1.Background = brush;
14}
15}
```

You need to ensure that the application changes the background color of the hosted control when the background color of the form changes.

Which code segment should you insert at line 05?

- A. elementHost1.PropertyMap.Remove("BackColor");
elementHost1.PropertyMap.Add("BackColor", new
PropertyTranslator(OnBackColorChange));
- B. elementHost1.PropertyMap.Remove("Background");
elementHost1.PropertyMap.Add("Background", new
PropertyTranslator(OnBackColorChange));
- C. elementHost1.PropertyMap.Add("BackColor", new
PropertyTranslator(OnBackColorChange));
elementHost1.PropertyMap.Apply("BackColor");
- D. elementHost1.PropertyMap.Add("Background", new
PropertyTranslator(OnBackColorChange));
elementHost1.PropertyMap.Apply("Background");

Answer: A

NO.9 You use Microsoft .NET Framework 4 to create a Windows Forms application.

You have a dataset as shown in the following exhibit.

You plan to add a DataGridView to display the dataset.

You need to ensure that the DataGridView meets the following requirements:

- Shows Order Details for the selected order.
- Shows only Order Details for items that have UnitPrice greater than 20.
- Sorts Products by ProductName

Which code segment should you use?

A. ordersBindingSource.DataSource = productsBindingSource;

ordersBindingSource.DataMember = "FK_Order_Details_Products";

productsBindingSource.Filter = "UnitPrice > 20";

productsBindingSource.Sort = "ProductName";

B. productsDataGridView.DataSource = ordersBindingSource;

productsBindingSource.Filter = "UnitPrice > 20";

productsBindingSource.Sort = "ProductName";

C. order_DetailsBindingSource.DataSource = ordersBindingSource;

order_DetailsBindingSource.DataMember = "FK_Order_Details_Orders";

order_DetailsBindingSource.Filter = "UnitPrice > 20";

productsBindingSource.Sort = "ProductName";

D. order_DetailsDataGridView.DataSource = ordersBindingSource;

order_DetailsBindingSource.Filter = "UnitPrice > 20";

productsBindingSource.Sort = "ProductName";

Answer: C

NO.10 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

The application contains a composite user control that includes a TextBox control named txtInput.

The user control will be hosted in a window and will have handlers for the text-changed event of txtInput.

You need to ensure that the application meets the following requirements:

AddHandler (TextBox.TextChangedEvent, new RoutedEventHandler (Audit_TextChanged), true);

Which of the following statements are TRUE? (choose all that apply)

A. A text-changed event handler, named Audit_TextChanged, was Created for the txtInput control.

- B. Audit_TextChanged will stop running because the event is marked as handled by certain event handlers.
- C. Even through the event is marked handled by certain event handlers, Audit_TextChanged will still run.
- D. Audit_TextChanged will continue to run until the event is marked as handled.
- Answer: AC

NO.11 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.)

```
01public class Contact
02{
03private string _contactName;
04
05public string ContactName {
06get { return _contactName; }
07set { _contactName = value; }
08}
09
10}
```

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>
```

The TextBox control is data-bound to an instance of the Contact class.

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. Replace line 01 with the following code segment. public class Contact : IDataErrorInfo

B. Replace line 01 with the following code segment. public class Contact : ValidationRule

C. Replace line 01 with the following code segment. public class Contact :

INotifyPropertyChanging

D. Add the following code segment at line 04. public event PropertyChangingEventHandler
PropertyChanging;

E. Modify line 07 with the following code segment:

```
set {
```

```
if (this.PropertyChanging != null)
```

```
PropertyChanging(this, new
```

```
PropertyChangingEventArgs("ContactName"));
```

```
if (string.IsNullOrEmpty(value))
```

```
throw new ApplicationException("Contact name is required");
```

```
_contactName = value;
```

```
}
```

F. Add the following code segment at line 09.

```
public string Error {
```

```
public string this[string columnName] {
```

```
get {
```

```
if (columnName == "ContactName" &&
```

```
string.IsNullOrEmpty(this.ContactName))
```

```
return "Contact name is required";
```

```
return null;
```

```
}
```

```
}
```

Answer: AF

NO.12 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

The application contains a composite user control that includes a TextBox control named txtInput.

The user control will be hosted in a window and will have handlers for the text-changed event of txtInput.

You need to ensure that the application meets the following requirements:

- Creates a text-changed event handler named Audit_TextChanged for the txtInput control.

- Executes Audit_TextChanged even when specific handlers mark the event as handled.

Which code segment should you add to the constructor of the user control.?

- A. `txtInput.TextChanged+=Audit_TextChanged;`
- B. `AddHandler (TextBox.TextChangedEvent, new RoutedEventHandler (Audit_TextChanged), true);`
- C. `EventManager.RegisterClassHandler (typeof (TextBox),TextBox.TextChangedEvent, new RoutedEventHandler (Audit_TextChanged), true);`
- D. `EventManager.RegisterClassHandler (typeof (TextBox), TextBox.TextChangedEvent, new RoutedEventHandler (Audit_TextChanged), false);`

Answer: B

NO.13 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF) application.

You write the following code segment.

```
public class Contact { private string _contactName;
public string ContactName {
get { return _contactName; }
set {
if (string.IsNullOrEmpty(value))
throw new ApplicationException("Contact name is required");
_contactName = value;
}
}
}
```

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

```
<TextBox Text="{Binding Path=ContactName, ValidatesOnExceptions=True,
UpdateSourceTrigger=PropertyChanged}" />
```

The TextBox control is data-bound to an instance of the Contact class. You plan to implement an

ErrorTemplate in the TextBox control.

You need to ensure that the validation error message is displayed next to the TextBox control. Which code

fragment should you use?

A. `<ControlTemplate>`

`<DockPanel>`

`<AdornedElementPlaceholder Name="box" />`

`<TextBlock Text="{Binding ElementName=box,`

`Path=AdornedElement.(Validation.Errors)[0].ErrorContent}" Foreground="Red" />`

`</DockPanel>`

</ControlTemplate>

B. <ControlTemplate>

<DockPanel>

<AdornedElementPlaceholder Name="box" />

<TextBlock Text="{Binding ElementName=box,
Path=(Validation.Errors)[0].Exception.Message}"

Foreground="Red" />

</DockPanel>

</ControlTemplate>

C. <ControlTemplate>

<DockPanel>

<ContentControl Name="box" />

<TextBlock Text="{Binding ElementName=box,
Path=ContentControl.(Validation.Errors)[0].ErrorContent}" Foreground="Red" />

</DockPanel>

</ControlTemplate>

D. <ControlTemplate>

<DockPanel> <ContentControl Name="box" />

<TextBlock Text="{Binding ElementName=box,
Path=(Validation.Errors)[0].Exception.Message}"

Foreground="Red" />

</DockPanel>

</ControlTemplate>

Answer: A

NO.14 You use Microsoft .NET Framework 4 to create a Windows Presentation Foundation (WPF)

application.

You plan to use an existing Windows Forms control named MyWinFormControl in the MyControls

assembly.

You need to ensure that the control can be used in your application.

What should you do?

A. Add the following code fragment to the application.

```
<Window x:Class="HostingWfInWpf.Window1"
```

```
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
```

```
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
```

```
xmlns:wf="clr-namespace:MyCompany.Controls;assembly=MyControls;"
```

```
Title="HostingWfInWpf" >  
<Grid>  
<ElementHost>  
<wf:MyWinFormControl x:Name="control" />  
</ElementHost>  
</Grid> </Window>
```

B. Add the following code fragment to the application.

```
<Window x:Class="HostingWfInWpf.Window1"  
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"  
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"  
xmlns:wf="clr-namespace:MyCompany.Controls;assembly=MyControls;"  
Title="HostingWfInWpf" >  
<Grid>  
<WindowsFormsHost>  
<wf:MyWinFormControl x:Name="control" />  
</WindowsFormsHost>  
</Grid> </Window>
```

C. Add the following code segment to the WindowsLoaded function.

```
ElementHost host = new ElementHost();  
host.Dock = DockStyle.Fill;  
MyWinFormControl control = new MyWinFormControl();  
host.Child = control;  
this.Controls.Add(host);
```

D. Add the following code segment to the WindowsLoaded function.

```
Grid grid = new Grid();  
System.Windows.Forms.Integration.WindowsFormsHost host = new  
System.Windows.Forms.Integration.WindowsFormsHost();  
MyWinFormControl control = new MyWinFormControl();  
grid.Children.Add(host);
```

Answer: B

NO.15 You use Microsoft .NET Framework 4 to create a Windows Forms application.

You add a new class named Customer to the application.

You select the Customer class to create a new object data source.

You add the following components to a Windows Form:

- A BindingSource component named customerBindingSource that is data-bound to the Customer object data source.

- A set of TextBox controls to display and edit the Customer object properties.
- Each TextBox control is data-bound to a property of the customerBindingSource component.
- An ErrorProvider component named errorProvider that validates the input values for each TextBox control.

You need to ensure that the input data for each TextBox control is automatically validated by using the ErrorProvider component.

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

- A. Implement the validation rules inside the Validating event handler of each TextBox control by throwing an exception when the value is invalid.
- B. Implement the validation rules inside the TextChanged event handler of each TextBox control by throwing an exception when the value is invalid.
- C. Implement the validation rules inside the setter of each property of the Customer class by throwing an exception when the value is invalid.
- D. Add the following code segment to the InitializeComponent method of the Windows Form.
`this.errorProvider.DataSource = this.customerBindingSource;`
- E. Add the following code segment to the InitializeComponent method of the Windows Form.
`this.errorProvider.DataSource = this.customerBindingSource.DataSource;`
`this.errorProvider.DataMember = this.customerBindingSource.DataMember;`

Answer: CD