

Q.In our WPF application we want to go for Event Tunneling implementation but confused with available snippets. Can you suggest which one we pick for implementation?

- 1). Button1_Click
- 2). MouseDoubleClick
- 3). **PreviewMouseDoubleClick**
- 4). RoutedMouseClick

Q. To apply styles for Label _____ or _____ should be specified.

- 1). **Style="{DynamicResource Stylename}"**
- 2). **Style="{StaticResource Stylename}"**
- 3). Style="{StaticResource: Stylename}"
- 4). All of the above

Q.You are developing a Windows Presentation Foundation (WPF) application with multiple windows. The majority of the buttons within the application are styled consistently. However, three buttons within the application must be styled differently. You need to ensure that the application can handle this styling requirement. What should you do?

- 1). Create the styles in the application resources. Specify the TargetType property to be Button for the default style but not for the three buttons.
- 2). Create the styles in the main window resources. Specify the TargetType property to be Button for the default style but not for the three buttons.
- 3). Create the styles in the main window resources. Omit the Key property for the default Button style and specify the Key property for the Button style and the three buttons.
- 4). **Create the styles in the application resources. Omit the Key property for the default Button style and specify the Key property for the Button style and the three buttons.**

Q. Which Statement is true about RoutedEvent?

- 1). Routed Event stores only the state of the control
- 2). Routed Event Stores only the Data when the event is fired
- 3). Routed Event stores the state and Data when the event is fired
- 4). **All the above**

Q. You are developing a Windows Presentation Foundation (WPF) application to display Categories in ListBox. You have created category class as follows:

```
public class Category
{
    public int ID {get; set; }
    public string Name { get; set; }
}
```

On Window load event you have written following code :

```
List<Category> categoryList = new List<Category>();
categoryList.Add(new Category(){ID=101, Name="Accessories"});
categoryList.Add(new Category(){ID=102, Name="Bikes"});
categoryList.Add(new Category(){ID=103, Name="Cars"});
lstCategory.DataContext = categoryList;
```

Which of the following is the correct representation for the ListBox lstCategory, so that it will display categories?

- 1). <ListBox name=lstCategory></ListBox>
- 2). <ListBox name=lstCategory Binding="Category"></ListBox>
- 3). **<ListBox name=lstCategory ItemSource={Binding}></ListBox>**
- 4). <ListBox name=lstCategory DataSource={Binding}></ListBox>

Q. Which of the statement is not true about CLR Property vs. Dependency Property?

- 1). **Both types of properties are used to initialize and retrieve values of data members**
- 2). Dependency Property can apply Styles which cannot be done with CLR property
- 3). Both Types of properties are used for databinding
- 4). All the above statements are false

Q. Which statements are True about Triggers in WPF ?

- 1)Triggers get active when a property lost a specified value.
 - 2)Triggers get active when a specified event is fired.
 - 3)Triggers get active when a binding expression reaches a specified value.
- 1). 1
 - 2). 2
 - 3). 3
 - 4). 1 & 3 Both
 - 5). **2 & 3 Both**

Q. We are developing a WPF application in which we are dealing with custom collection. If there are any change in custom collection we have to send some notification to our

UI elements which is bound with Collection.

What should be used in the code to achieve this behavior?

- 1). ICollectionChanged
- 2). INotifyChanged
- 3). **INotifyCollectionChanged**
- 4). INotifyPropertyChanged
- 5). All of the above

Q. We are going through a requirement in one of our WPF application where we have to fill the background of Button control with more than one colour combination.
The brush object I can use for this is _____.

Attempted

1. LinearGradientBrush

Solution

1. LinearGradientBrush

2. SolidBrush

3. ImageBrush

4. GeometryObject

Q. You are developing a Windows Presentation Foundation (WPF) application.
You have a class named Orders, which contains OrderId and ShipOn properties.
You place a control in the MainWindow.xaml file, as follows.

```
<ComboBox x:Name="cmbOrders"
           ItemSource="{Binding Orders}">
</ComboBox>
```

When you run the application, the ComboBox control displays the class name for every row.
You need to ensure that the ComboBox control displays the OrderId and ShipOn values in columns.
Which markup segment should you add at line 03?

1). <ItemsControl.ItemsPanel>
<ItemsPanelTemplate>
<Grid>
<Grid.ColumnDefinitions>
<ColumnDefinition />
<ColumnDefinition />
</Grid.ColumnDefinitions>
<TextBlock Grid.Column="0" Text="{Binding ShipOn}"/>
<TextBlock Grid.Column="0" Text="{Binding OrderId}"/> </Grid>

```
</ItemsPanelTemplate>
</ItemsControl.ItemsPanel>
```

2). <ItemsControl.ItemTemplate>

```
<DataTemplate>
<Grid>
<TextBlock Text="{Binding OrderId}"/>
<TextBlock Text="{Binding ShipOn}"/>
</Grid>
</DataTemplate>
</ItemsControl.ItemTemplate>
```

3). <ItemsControl.ItemTemplate>

```
<DataTemplate>
<Grid>
<Grid.ColumnDefinitions>
<ColumnDefinition />
<ColumnDefinition />
</Grid.ColumnDefinitions>
<TextBlock Grid.Column="0" Text="{Binding OrderId}"/>
<TextBlock Grid.Column="1" Text="{Binding ShipOn}"/> </Grid>
</DataTemplate>
</ItemsControl.ItemTemplate>
```

4). <ItemsControl.ItemsPanel>

```
<ItemsPanelTemplate>
<Grid>
<TextBlock Text="{Binding OrderId}"/>
<TextBlock Text="{Binding ShipOn}"/>
</Grid>
</ItemsPanelTemplate>
</ItemsControl.ItemsPanel>
```