



```

22         </MultiBinding>
23     </CheckBox.IsChecked>
24 </CheckBox>
25 </DataTemplate>
26 </ListBox.ItemTemplate>
27 </ListBox>

```

Line 4: We specified SelectionMode as Multiple, so we can choose multiple checkboxes.

Line 5, 6, 9: These lines are used to make **ListBox** style as WrapPanel.

Line 18: It is necessary to get selected checkboxes. To get SelectedItems: `var selectedCheckBoxList = lstCountries.SelectedItems;`

Line 19: `Path="Active"` is necessary when initialize **CheckBox** items.

#### Step 4: CollectionToBoolConverter Class

```

1  public class CollectionToBoolConverter : IMultiValueConverter
2  {
3      /// <summary>
4      /// Runs when initialization
5      /// </summary>
6      /// <param name="values">Binding values</param>
7      /// <param name="targetType">Merged value</param>
8      /// <param name="parameter"></param>
9      /// <param name="culture"></param>
10     /// <returns></returns>
11     public object Convert(object[] values, Type targetType, object parameter,
12         CultureInfo culture)
13     {
14         bool listBoxValue = (bool) values[0];
15         bool initializedValue = (bool) values[1];
16         return listBoxValue || initializedValue;
17     }
18     /// <summary>
19     /// Runs when any ListBox element that's CheckBox element is clicked.
20     /// This method only runs when Mode is "TwoWay"
21     /// </summary>
22     /// <param name="value"></param>
23     /// <param name="targetTypes"></param>
24     /// <param name="parameter"></param>
25     /// <param name="culture"></param>
26     /// <returns></returns>
27     public object[] ConvertBack(object value, Type[] targetTypes, object parameter,
28         CultureInfo culture)
29     {
30         /*bool isChecked = (bool) value;
31         object[] resultObjects=new object[2];
32         resultObjects[0] = isChecked;
33         resultObjects[1] = isChecked;
34         return resultObjects;
35         */
36         return null;
37     }
38 }

```