

Bounded to ComboBox



Birthday

Back

Forward

```
//File:Window.xaml.cs
using System;
using System.Collections.Generic;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
```

```
using System.Windows.Shapes;
using System.Collections.ObjectModel;
using System.ComponentModel;

namespace WpfApplication1 {

public class Person : INotifyPropertyChanged {
public event PropertyChangedEventHandler PropertyChanged;
protected void Notify(string propName) {
if( this.PropertyChanged != null ) {
PropertyChanged(this, new PropertyChangedEventArgs(propName));
}
}

string name;
public string Name {
get { return this.name; }
set {
this.name = value;
Notify("Name");
}
}

int age;
public int Age {
get { return this.age; }
set {
this.age = value;
Notify("Age");
}
}
}

class People : ObservableCollection { }

public class NamedAge : INotifyPropertyChanged {
public event PropertyChangedEventHandler PropertyChanged;
protected void Notify(string propNameForAge) {
if( this.PropertyChanged != null ) {
```

```

PropertyChanged(this,
PropertyChangedEventArgs(propNameForAge));
}
}

string nameForAge;
public string NameForAge {
get { return this.nameForAge; }
set {
this.nameForAge = value;
Notify("NameForAge");
}
}

int ageId;
public int AgeId {
get { return this.ageId; }
set {
this.ageId = value;
Notify("AgeId");
}
}
}

class NamedAges : ObservableCollection { }

public partial class Window1 : System.Windows.Window {

public Window1() {
InitializeComponent();
lb.MouseDoubleClick += lb_MouseDoubleClick;
this.birthdayButton.Click += birthdayButton_Click;
this.backButton.Click += backButton_Click;
this.forwardButton.Click += forwardButton_Click;
}

void lb_MouseDoubleClick(object sender, MouseButtonEventArgs
e) {
int index = lb.SelectedIndex;

```

```
if( index < 0 ) { return; } Person item =
(Person)lb.SelectedItem; int value = (int)lb.SelectedValue;
Console.WriteLine(index); Console.WriteLine(item.Name);
Console.WriteLine(item.Age); Console.WriteLine(value); }
ICollectionView GetFamilyView() { People people =
(People)this.FindResource("Family"); return
CollectionViewSource.DefaultView(peoples); } void
birthdayButton_Click(object sender, RoutedEventArgs e) {
ICollectionView view = GetFamilyView(); Person person =
(Person)view.CurrentItem; ++person.Age;
Console.WriteLine(person.Name); Console.WriteLine(person.Age);
} void backButton_Click(object sender, RoutedEventArgs e) {
ICollectionView view = GetFamilyView();
view.MoveCurrentToPrevious(); if( view.IsCurrentBeforeFirst )
{ view.MoveCurrentToFirst(); } } void
forwardButton_Click(object sender, RoutedEventArgs e) {
ICollectionView view = GetFamilyView();
view.MoveCurrentToNext(); if( view.IsCurrentAfterLast ) {
view.MoveCurrentToLast(); } } } } [/csharp]
```