Worksheet #3

Overview

You should have Visual Studio 2012 open.

1. Create a new Project either from the shortcut on the Start page or through the file menu,
2. Select the type to be a C# WPF Application.
3. In the Location textbox, enter your desired hard drive location. Note that this will be on the mounted z: drive in the CSE environment. Name the Project *Lastname*Fav5Web where *Lastname* is replaced with your name or an alias.
4. Select OK.
5. If the solution explorer is not open, open it with View->Solution Explorer or the shortcut Ctrl-W,S. You will see tree views for the Properties, References and App.xaml and MainWindow.xaml. These last two files use XML to describe the design of your application. The compiler automatically generates source code from these.
6. Expand the MainWindow.xaml file. You will see a MainWindow.xaml.cs file. Click on this file. This opens the MainWindow.xaml.cs file in the regular source code editor. Note that this class has a new keyword **partial** in its definition. This indicates to the compiler that this is only part of the class definition and the rest of the class is defined elsewhere (in the .xaml file).
7. To see how this works, click "Start" to build and start your application now. You will see a blank window! Close your application for now and browse to your project directory using the windows explorer. Now open the obj/Debug (or obj/Release) directory. You will see some classes MainWindow.g.cs and MainWindow.g.i.cs. Go ahead and examine these. These were generated by the compiler and contain the rest of the definition of your MainWindow class. You should not edit these files directly.
8. The proper way to think of these separate files with partial class definitions is that the compiler will merge the sources together and then compile them. In general you can have several files defining a class as long as they are all available during compilation.

Create MyFavFiveWeb application

We are going to create a simple web browser with a constrained set of predetermined web sites to visit as illustrated to the right. It has a large pane to view the web page with a combo box and button below to select the web page to display.

11. Right-click on MainWindow.xaml in the Solution Explorer and select Rename. Change MainWindow to MyBrowser. Note that it renamed the files only, but not the class name.
12. Click on MyBrowser.xaml.cs to bring up the source code view. In the source code, right-click on MainWindow and select Refactor->rename. Change the name to MyBrowser. You will be asked to confirm these changes throughout the code.
13. Finally, click on App.xaml. Change the line that sais "StartupUri="MainWindow.xaml" to point to MyBrowser.xaml. Go ahead and run your application now to test you don't have any errors.
14. Note that the window still sais MainWindow. Lets change this now. Click on the window In MyBrowser.xaml. Change the title property to be something more meaningful. (If you don't see properties in the lower right corner, click View → Properties Window.)
15. Note: With xaml, you could have also changed the title just by editing the xml in the bottom half of the design screen.

```
<Window x:Class="worksheet3.MyBrowser"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="David's Favorites Browser" Height="350" Width="525">
    <Grid/>
</Window>
```

16. Go ahead and change the size of your window to 800 by 1024.
17. Review the other properties to become familiar with them.
18. With the Designer and the XAML, you have two methods of designing your interface. Lets go ahead and work with the XML to add two rows to your window: a fixed row on the bottom for controls and an expanding row above for a web browser. Add the following code to your MyBrowser.xaml in place of the <Grid/> element:

```
<Grid>
    <Grid.RowDefinitions>
        <RowDefinition Height="*"/>
        <RowDefinition Height="80"/>
    </Grid.RowDefinitions>
</Grid>
```

Notice as you type the text, intellisense help is available for you to offer appropriate choices.

19. Now let us use the visual designer. From the All WPF Controls of the Toolbox drag a WebBrowser control to the upper panel.
20. Drag a Button to the bottom panel.
21. Rename the button **(Name)** property to *buttonGo*. Set the **Content** to *Go.*
22. *S*et the **Horizontal Alignment** Property to the *Right*. And the right margin to 30.
23. Set the **Vertical Alignment** property to Center and the top and bottom margins to 0.
24. Drag a ComboBox to the bottom panel.
25. Play with the layout settings to get the combo box to align with the button and take up most of the space to the left of it.
26. Name your combo box comboFavoriteLinks.
27. On the Items Property (of the ComboBox), click the little … button. This will bring up a Collection Editor. On the bottom of the collection editor, switch the dropdown that says “Button” to “ComboBoxItem”, then click add.
28. Change the Content property to be one of your favorite web addresses.
29. Add 4 more of your favorite websites as combo box items.
30. Add a web control to the top grid row.
31. Name your web control controlWebBrowser.
32. Set your web control horizontal and vertical alignment to “stretch” and margins to 0. Set height and width to Auto by clicking the icon next to the input field.
33. Build and Run your application. You will note that the combo box items can be selected. The Go button does nothing, and no web page is displayed. We now have a working GUI, but no logic.
34. Double-click the Go button. This will take you over to your MyBrowser.xaml.cs code editor. A new method has been added as so:

```
private void buttonGo_Click(object sender, RoutedEventArgs e)
{
```

```
}
```

31. Add the following line to this new method:

```
controlWebBrowser.Navigate((comboFavoriteLinks.SelectedItem as
ComboBoxItem).Content.ToString());
```

Note that SelectedItem gives us an Object, so we cast it to a ComboBoxItem. Content is also an object, so we call its ToString method.

32. Build and Run your application. Selecting an item in the combo box and pressing Go now works! Magic!