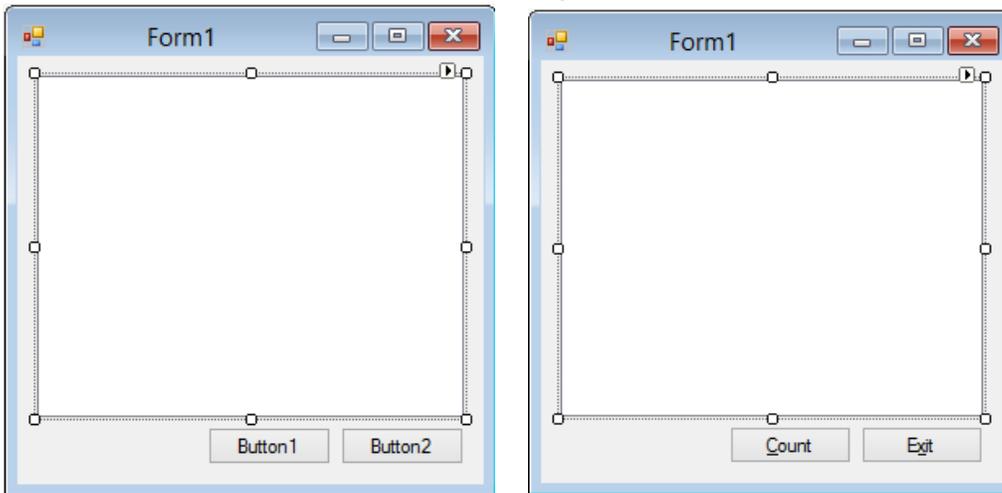# Renaming a Solution and Its Directories

This little project will help you learn more about the file structure of Visual Basic and how to rename a project and its directories.

The sample project

1. Start a new project with the default name of WindowsApplication1.
2. Add a ListView and two buttons from the Toolbox to the form.
3. Position them in the form similar to the image on the left:
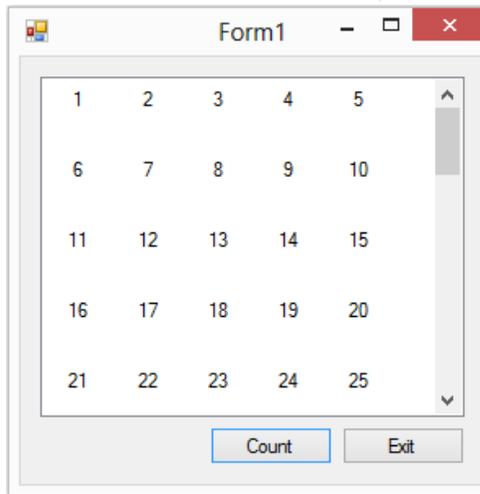

Before renaming   **After renaming**

4. Rename the left button to btnCount with the Text property set to &Count.
5. Rename the right button to btnExit with the Text property set to E&xit.
6. Rename the ListView to ltvCount.
7. Double-click on the Count button and enter the following code:

```vb
Private Sub btnCount_Click(…) Handles btnCount.Click
        ltvCount.Clear()
        For X As Integer = 1 To 100
            ltvCount.Items.Add(X.ToString)
        Next
End Sub
```
8. Double-click the Exit button and enter the following code:
```vb
Application.Exit()
```

9. When you run the project and select the Count button you should see:

10. Select the Exit button to terminate the program.

The renaming process

11. Notice that the form name is still Form1 and the project name is WindowsApplication1.
12. Single-click on Form1.vb in the Solution Explorer. Pause long enough that the computer knows you're not double-clicking and single-click on Form1.vb again. Form1 should be selected or highlighted without the .vb.
13. With Form1 highlighted change the name to frmMyBigCounter (no spaces).
14. Right-click on the project name, WindowsApplication1, and rename it to My Big Counter (with the spaces).
15. Right-click on the project name in the Solution Explorer and select Properties at the bottom of the context menu and it will display the project properties window.
16. On the Application tab of the project properties window, change the name of the Assembly name to the new name of your project.
17. Change the Root namespace to the name of your new project. The Root namespace cannot contain spaces so if your new project name contains spaces remove them in the Root namespace or replace them with underscore characters (_). Example: A project renamed from WindowsApplication1 to My Big Counter should be MyBigCounter or My_Big_Counter in the Root namespace box.
18. Make sure the Startup form dropdown box contains the correct form name.
19. Select the Assembly Information button. You will see two instances of the original name. Change it to the new name with spaces.
20. Save All and run the project just to make sure it still works.
21. Close the Visual Studio.

Renaming the directories

22. Using Windows Explorer, examine the directory where your project is stored. (Usually the directory is: C:\Users\UserName\Documents\visual studio 2012\Projects.)
23. Locate the directory that bears the name of the original project and change it to the new project name.
24. Now select the newly renamed directory to see the files and directories it contains.

25. Inside that directory you may see another directory of the original project name. If so, change it to the new project name.
26. Open Notepad.
27. Drag the file in the current directory with the .sln extension to into Notepad. (This is the solution file ".sln".)
28. Locate the original project name in the path in the solution file and change it to the new project name.
29. After saving the .sln file double-click on it to start the project.
30. You should be able to run the project without exceptions.

Deleting the old files

31. If you look in the directory that contains the .sln file you will see one with the new name and one with the old. The same is true for the .suo file. You can delete all files in this directory that contain the old name.
32. Select the next level directory with the new name. Select the bin directory and then the Debug directory. This is called the output directory because this is where VB puts the files it creates when you run the project. Notice the files with the new name and compare to those with the old name. You can delete all of the files with the old name. Actually, you can delete all of the files in this directory because VB will recreate them the next time you run your project.

The Bottom Line

This works but it's a lot of work and the real solution is to name your projects correctly when you start.