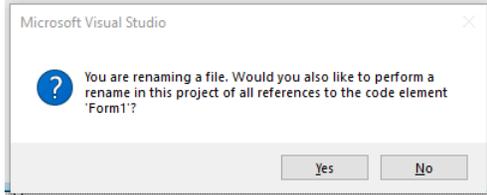
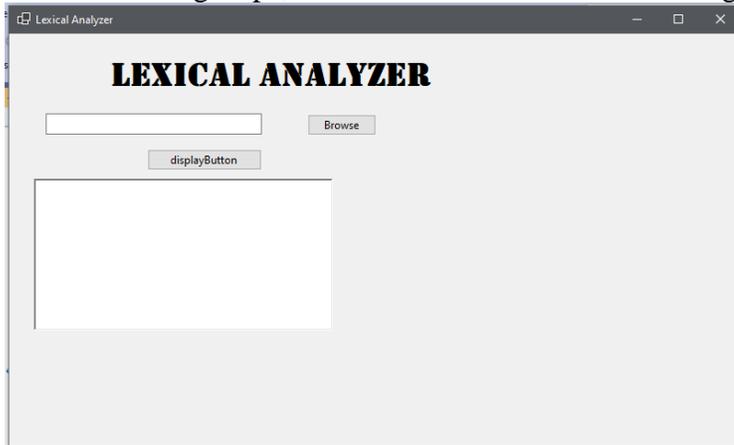


Concepts of Programming Languages, CSCI 305, Fall 2021 Lab 4, C#, Oct. 8

1. Create a C# WinForm application.
2. In the Solution Explorer (probably on the right) give the form file a better name than “Form1.cs”. I’ll use Main.cs. Let this change permeate to the other references by clicking “Yes” on the following popup box.



3. Give the title on the form a better name by changing its text property. (Select the form and find the “Text” property in the Properties window which is probably located below the Solution Explorer.)
4. In the following steps, create a form that looks something like the following:



5. Add a label to the form by dragging a label from the Toolbox (probably on the left when a form is selected) and dropping it onto the form. Make its text property “Lexical Analyzer” and choose a larger font so this can be a title of your form. You can just leave the name label1, since the code won’t be making any changes to this label.
6. Add a text box to hold the name of the code file to analyzed, and give the text box a reasonable name. (I’ll call the text box, fileNameTextBox.)
7. Add a button to the right of the text box (I’ll use browseButton) and set its text property to “Browse”.
8. Double click on the button and the skeleton of a function is added to the .cs file. This is where you can add the code which will execute when the button is clicked. (Careful, if you do this twice, skeleton code will be generated twice and you’ll need to remove the extra code. In addition to removing the skeleton code, you also need to remove the code for the listener which was added to the designer code that we ordinarily don’t modify. You will have to remove it from there too. It will probably be the last line in the code associated with the object.)

9. Add the code within the braces to the skeletal function to create an OpenFileDialog to facilitate getting the name of a file holding the code to be analyzed. (<https://www.c-sharpcorner.com/UploadFile/mahesh/openfiledialog-in-C-Sharp/>)

```
private void browseButton_Click(object sender, EventArgs e)
{
    OpenFileDialog openFileDialog1 = new OpenFileDialog
    {
        InitialDirectory = @"C:\",
        Title = "Browse Text Files",

        CheckFileExists = true,
        CheckPathExists = true,
        DefaultExt = ".txt",
        Filter = "txt files (*.txt)|*.txt",
        FilterIndex = 2,
        RestoreDirectory = true,

        ReadOnlyChecked = true,
        ShowReadOnly = true
    };

    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        fileNameTextBox.Text = openFileDialog1.FileName;
    }
}
```

10. Add a rich text box to hold the code to be analyzed and give this a good name as you will be using this name a lot. (I'll use codeDisplayRichTextBox.)
11. Add another button to open the file and display its contents in the rich text box using the code (this requires the System.IO library so add "using System.IO" to top of Main.cs)

```
private void xxxxButton_Click(object sender, EventArgs e)
{
    codeDisplayRichTextBox.Text = File.ReadAllText(fileNameTextBox .Text);
};
```

The tutorial <https://www.c-sharpcorner.com/article/search-and-highlight-text-in-rich-textbox/> tells how to highlight text within the rich text box.

