

Laboratory Exercise

Reading a File

Objectives:

At the end of the exercise, the students should be able to:

- Write programs that correctly handle exceptions; and
- Create customized exceptions.

Software Requirement:

- Visual Studio IDE 2015 or higher

Procedures:

1. Get your previous program (**05 Laboratory Exercise 1**) from your instructor. This is a continuation of your previous laboratory exercise.
2. Do not open the previous activity. Create a separate program that opens a text file from a specific folder. This program only contains a single form named FrmOpenFile. See Figure 1.

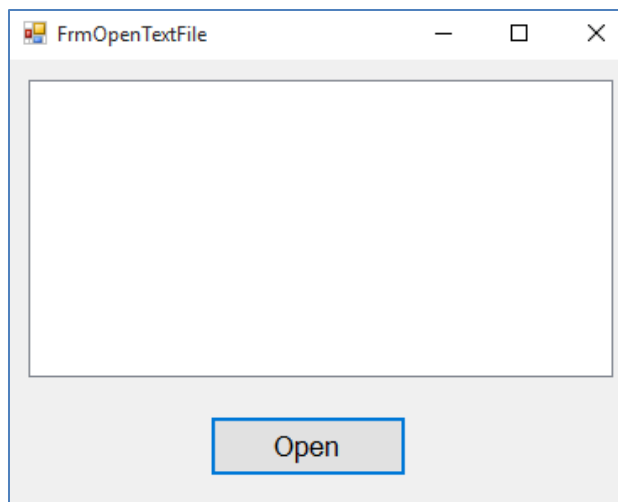


Figure 1. FrmOpenTextFile

3. Follow the given design in Figure 1. See Table 1 for the names of the control that contains function.

Forms/Controls	Name	
Form	FrmOpenTextFile	

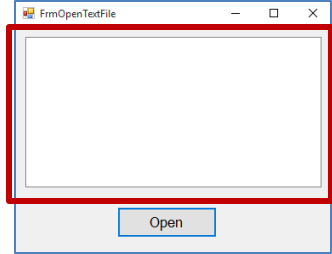
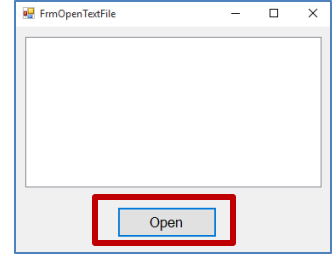
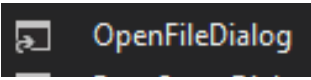
Forms/Controls	Name	
ListView	lvShowText	
Button	btnOpen	

Table 1. Forms and controls

- In the Toolbox, look for **OpenFileDialog**  and drag to the **FrmOpenTextFile**.
- Click the **Open** button to generate a click event.
- Create a method named **DisplayToList** and add the following code:

```

openFileDialog1.InitialDirectory = @"C:\";
openFileDialog1.Title = "Browse Text Files";
openFileDialog1.DefaultExt = "txt";
openFileDialog1.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";
openFileDialog1.ShowDialog();
path = openFileDialog1.FileName;
using (StreamReader streamReader = File.OpenText(path)) {
    string _getText = "";
    while ((_getText = streamReader.ReadLine()) != null) {
        Console.WriteLine(_getText);
        lvShowText.Items.Add(_getText);
    }
}

```

- Run the program to check the output and analyze the given code.

Challenge Exercise:

- In this challenge exercise, open your previous project and another form named **FrmStudentRecord**.

2. Add another button to the FrmRegistration to open the form FrmStudentRecord.

3. The **Register** button in FrmStudent should go back to the FrmRegistration. The **Find** button will look for the text file in the specified directory, while the **Upload** button will only display a message box "Successfully Uploaded!" and will clear the ListView's existing content.
4. Run the program to check the output.
5. Inform your instructor once you're done with the Challenge Exercise as this will be recorded.

Grading Rubric:

CRITERIA	PERFORMANCE INDICATORS	POINTS
Correctness	The code produces the expected result.	30
Logic	The code meets the specifications of the problem.	30
Efficiency	The code is concise without sacrificing correctness and logic.	20
Syntax	The code adheres to the rules of the programming language.	20
Total		100