Object Oriented Programming

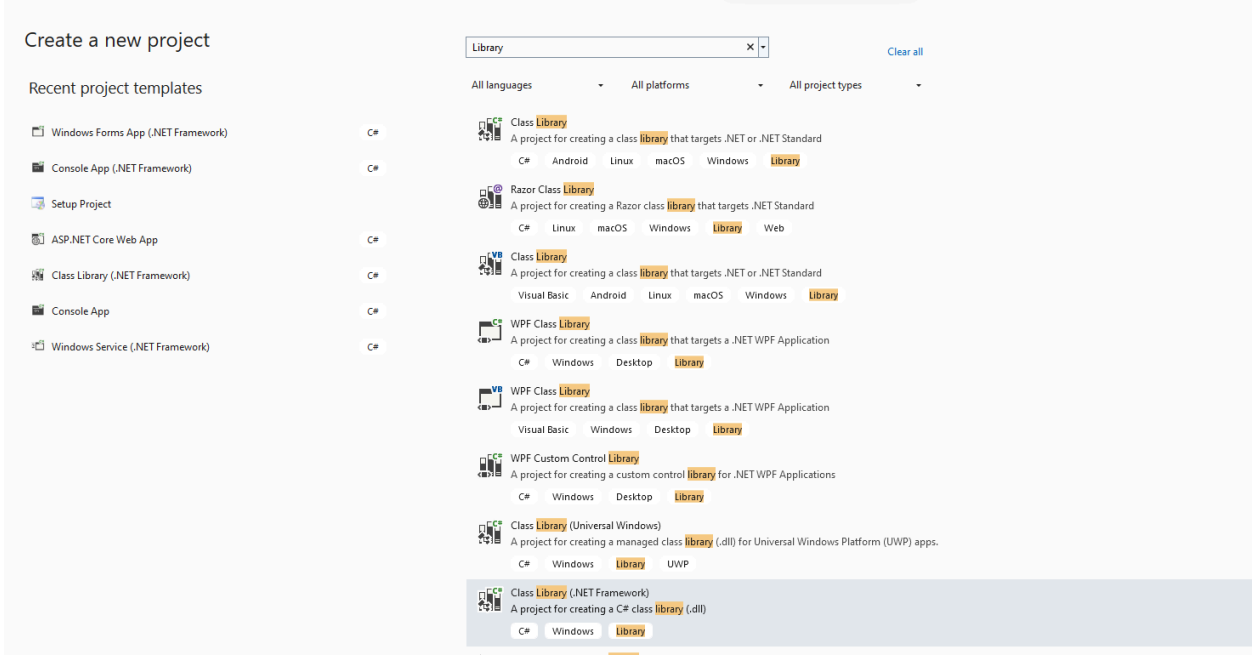
Lab Manual 09

Introduction

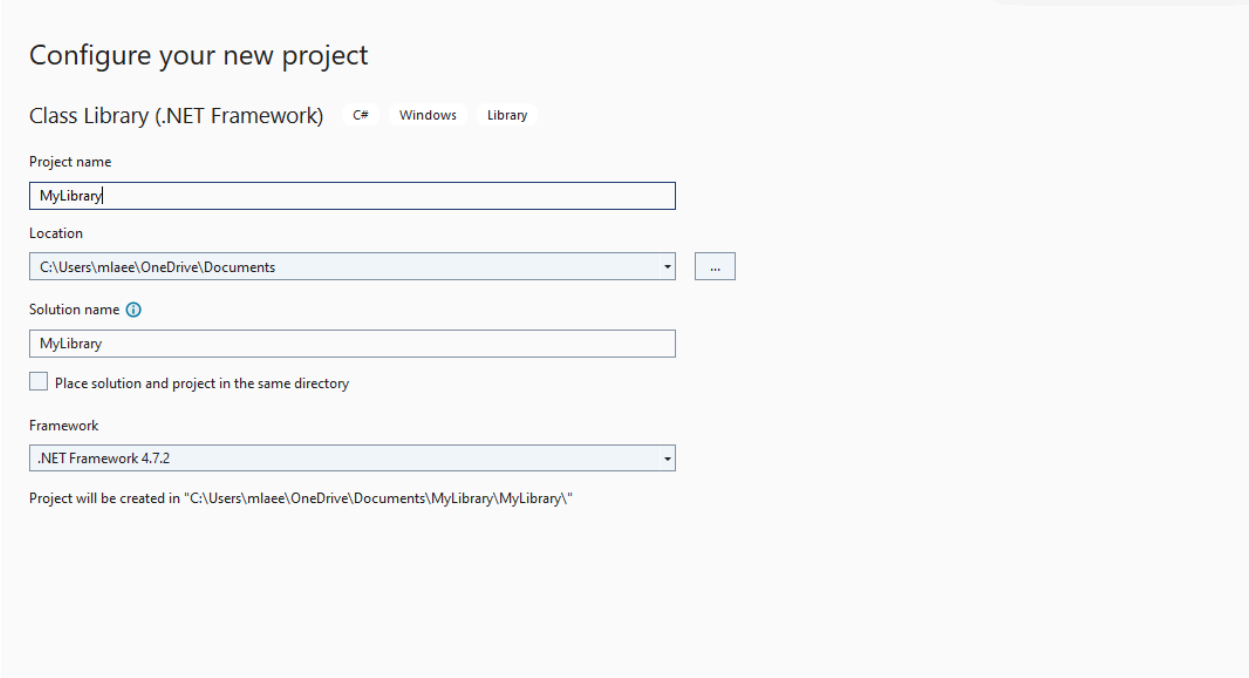
Welcome back Students.

This lab manual is going to be very important as it will teach you how to write the best reusable code in your project. As you know, we are using Business Layer (BL), Data Layer (DL), and User Interface (UI) layers in our object-oriented programming.Now, we will separate our DL and BL layers, as well as other supporting files like utilities (Utils), from the existing project. We will then create a new project to generate a DLL file. This DLL file can be included in any project because we will now use the same DLL file with both Console-based applications and Windows Form-based applications, and it should work seamlessly.

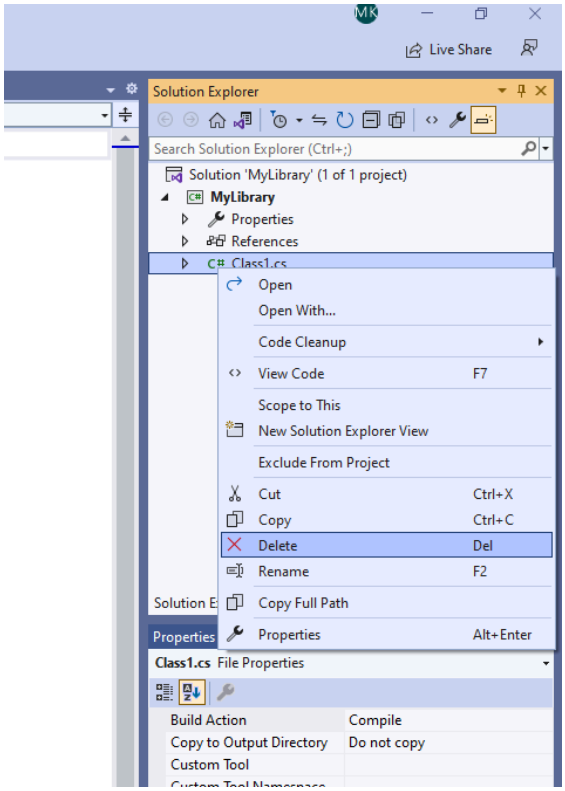
Step 1: Creating your Library Project in your Visual Studio
Select **Class Library (.NET Framework)** project .



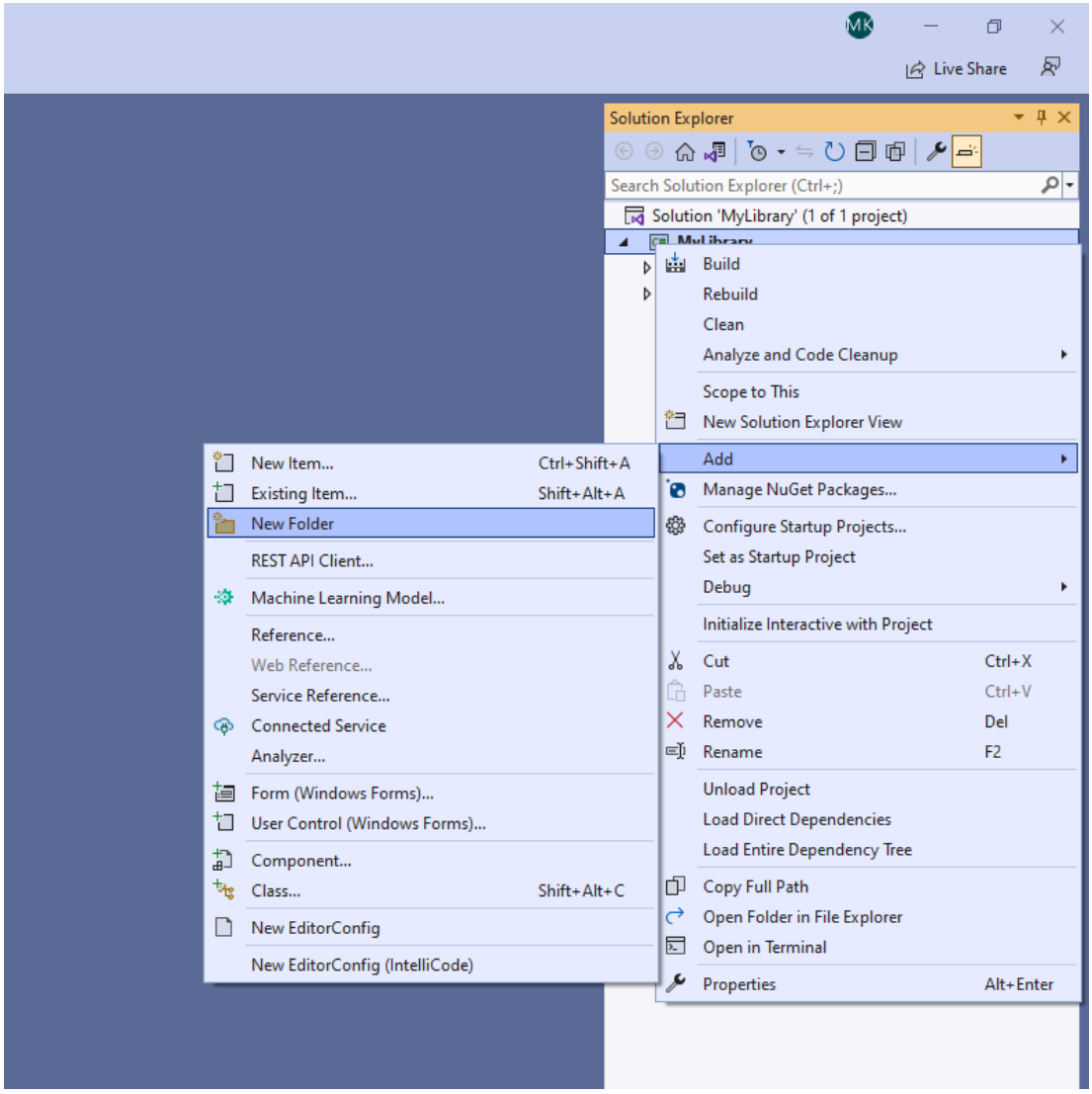
Give any meaningful project name to your library.



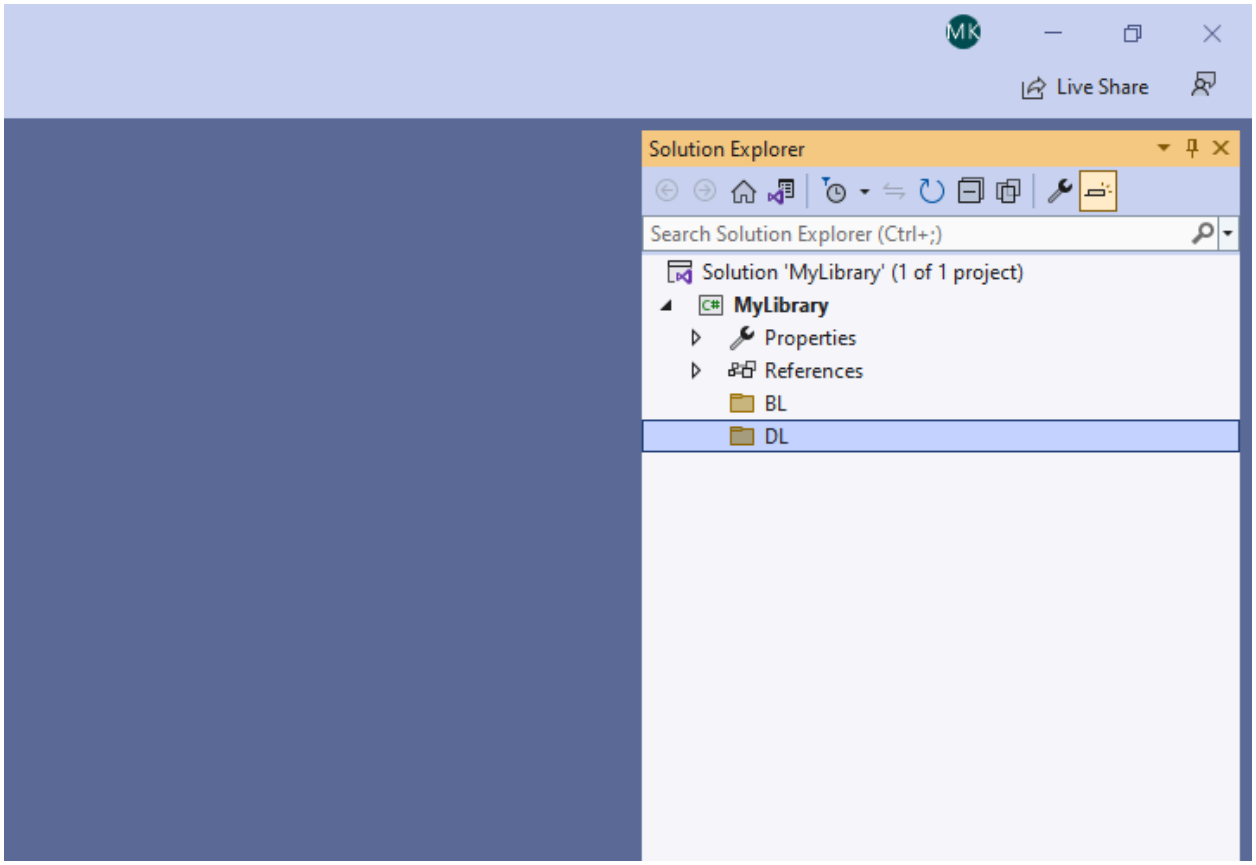
Delete your default generated code



Create BL and DL folders in your new project

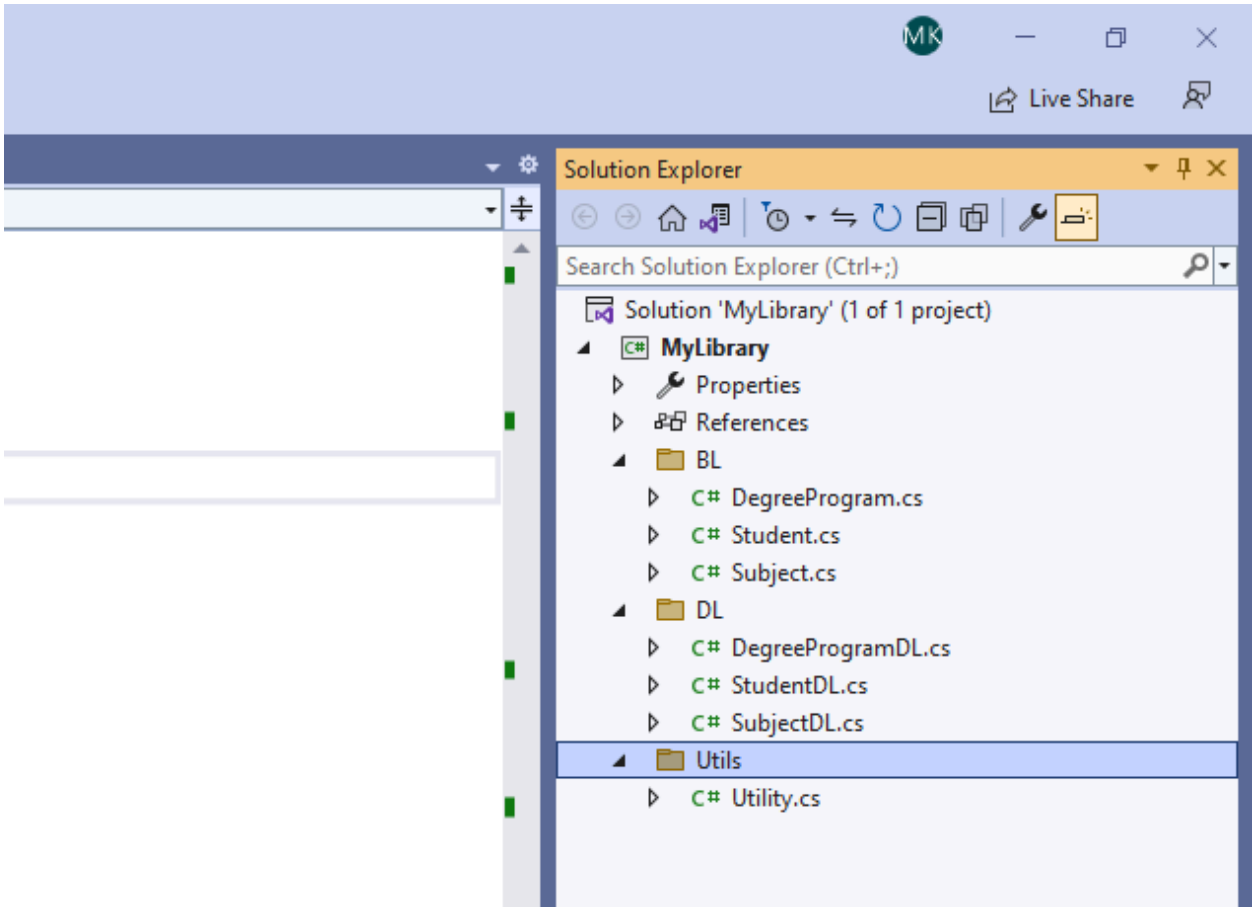


It should look like this



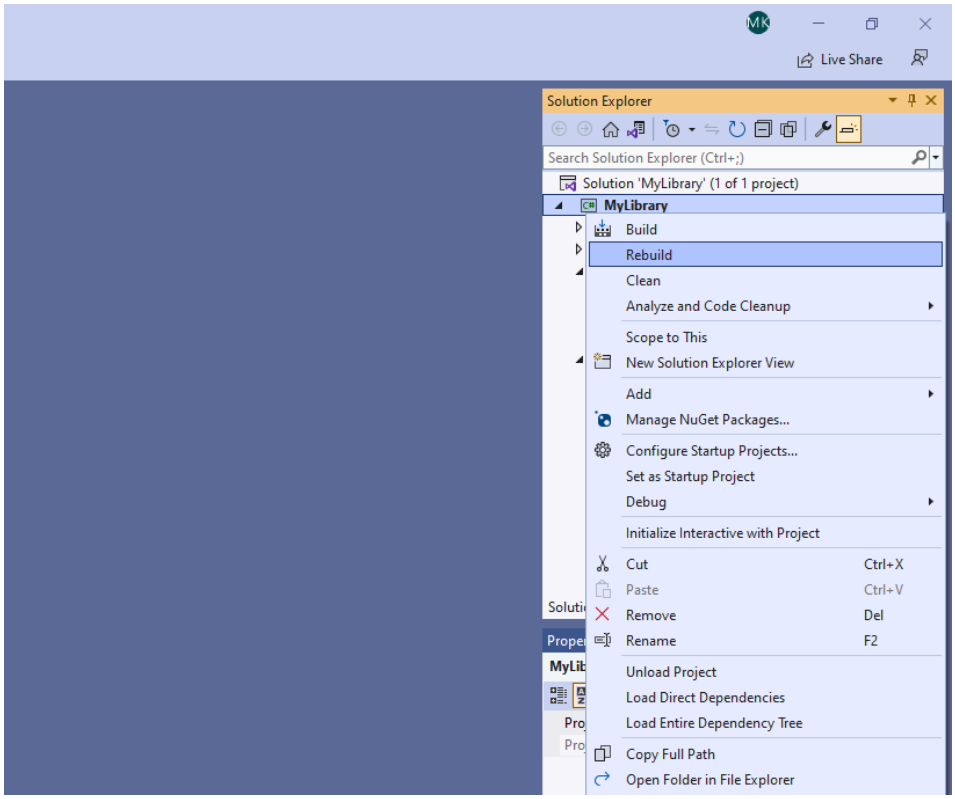
Copy the existing BL and DL files from your project into these folders. If you don't have any existing code, you can create the classes in this project based on your previous knowledge. However, keep in mind that this project will not execute any code and will not have a main function.

After copying the code files into this project, it should resemble the following structure, or your project will contain your files:

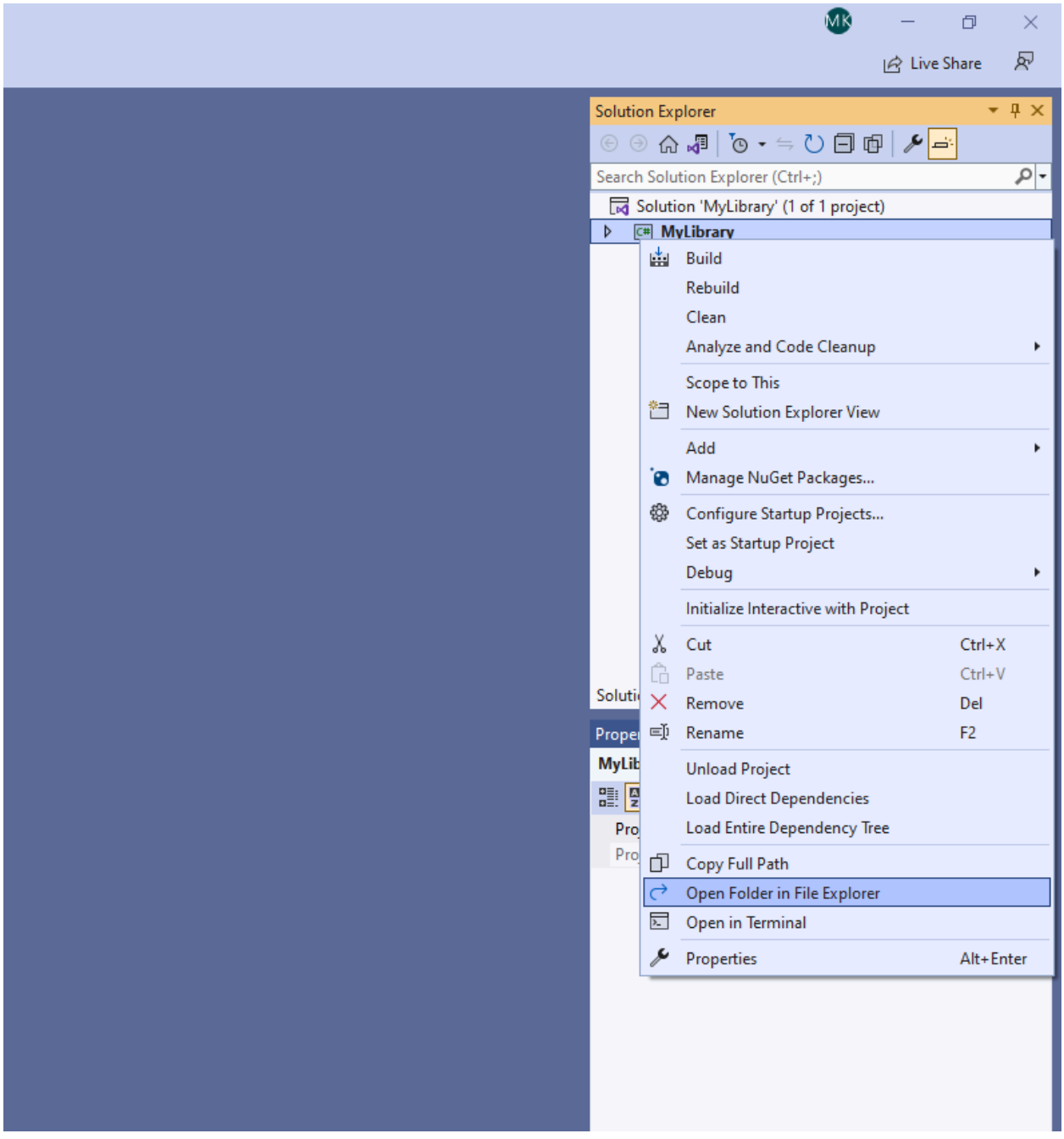


Note: The Utility class is simply a class containing some common functions, such as returning the database connection string. You can choose your preferred method for using the connection string, but it's advisable to retrieve it from a function. This way, you can easily modify the function later to read the connection string from another source, if needed.

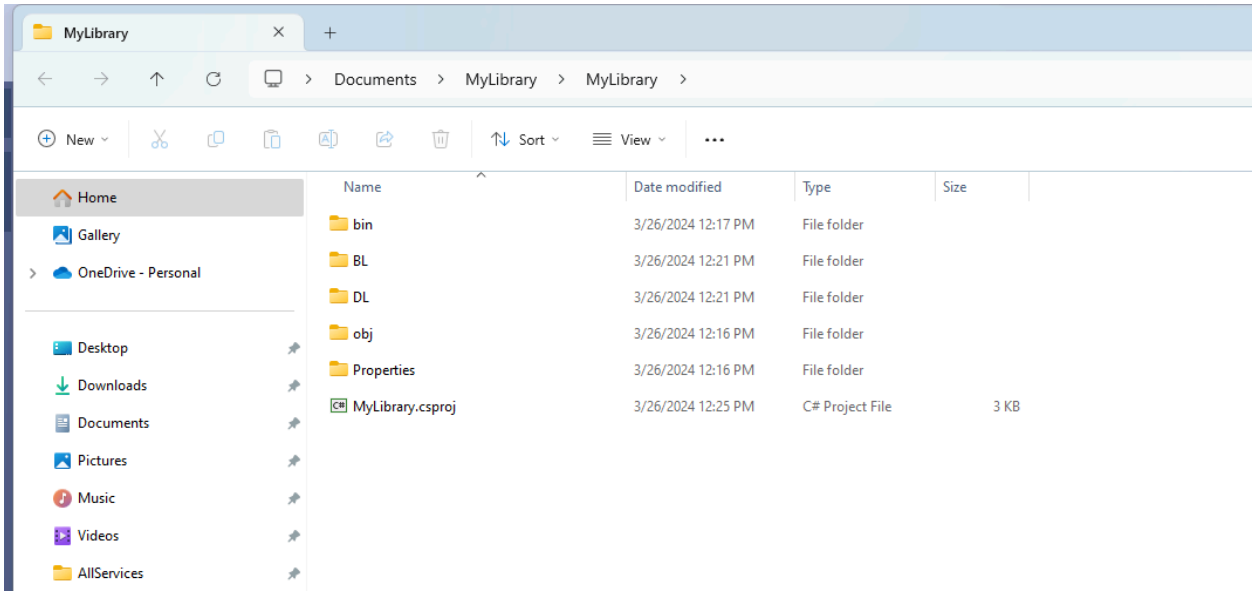
Right-click on your project name and use the 'Build' or 'Rebuild' option. Make sure your project is built successfully without any errors or failures; otherwise, it will not generate the DLL files.



You can locate your library project by right-clicking on the library project and selecting 'Show in Explorer'



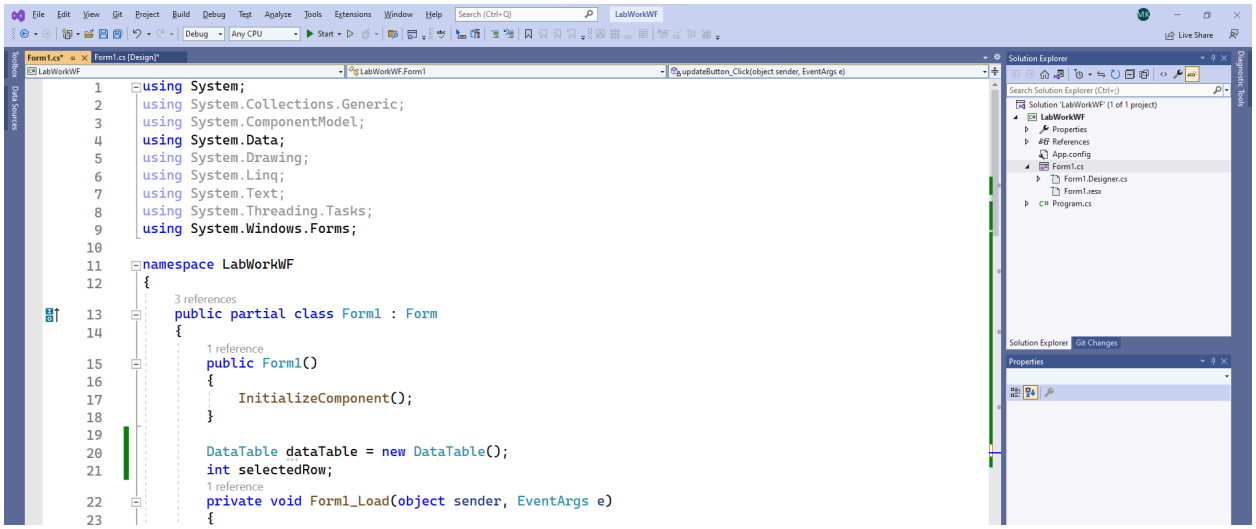
Copy the Address



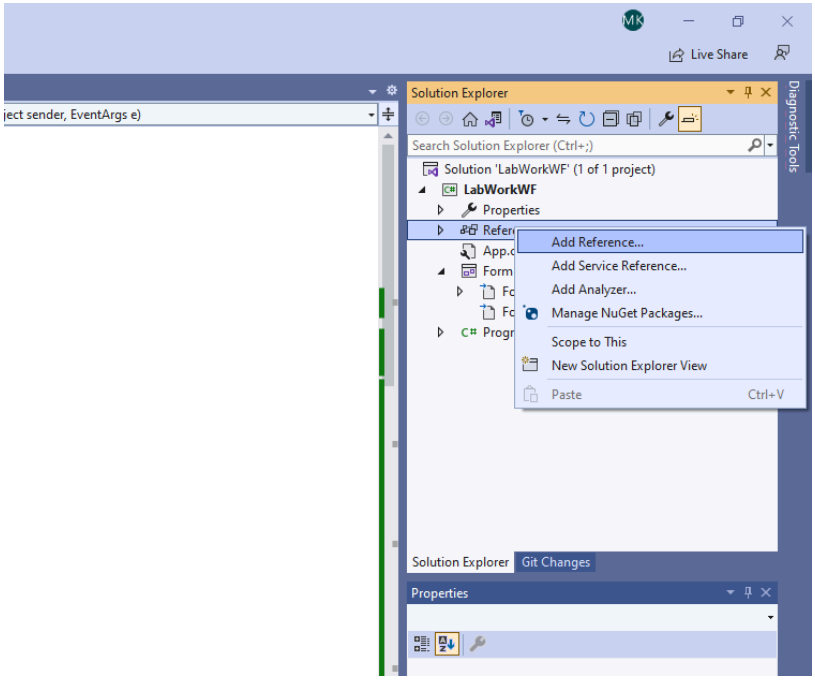
Copy the address of this project, or you can remember it for a short time because in the next step, we will use this path to add this project to our new Windows Form or Console-based application project.

Now, open your existing Windows Form or Console-based project and add the Library project to your solution

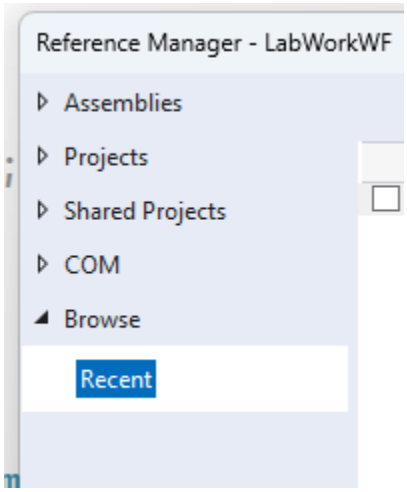
Here is my project



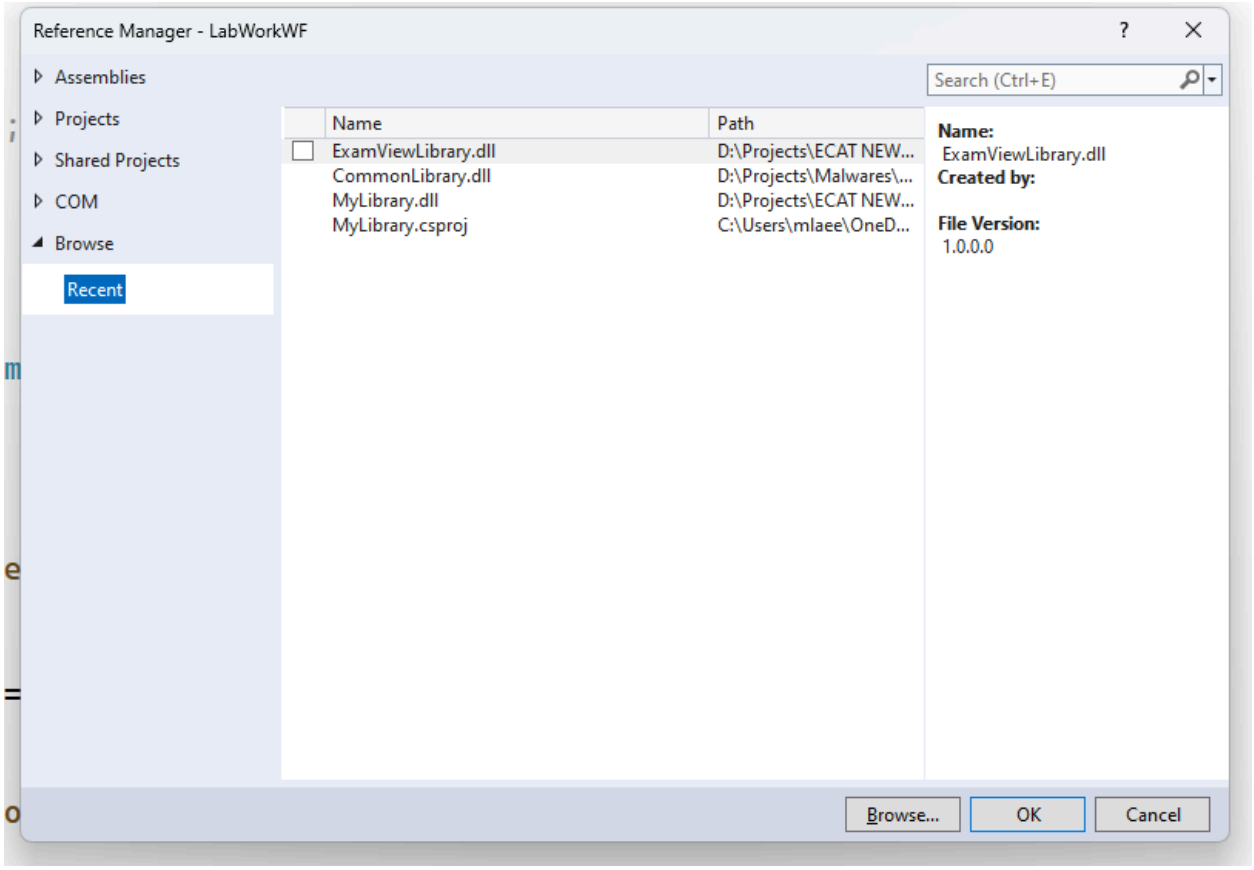
Right click on the References option in your Solution Explorer.



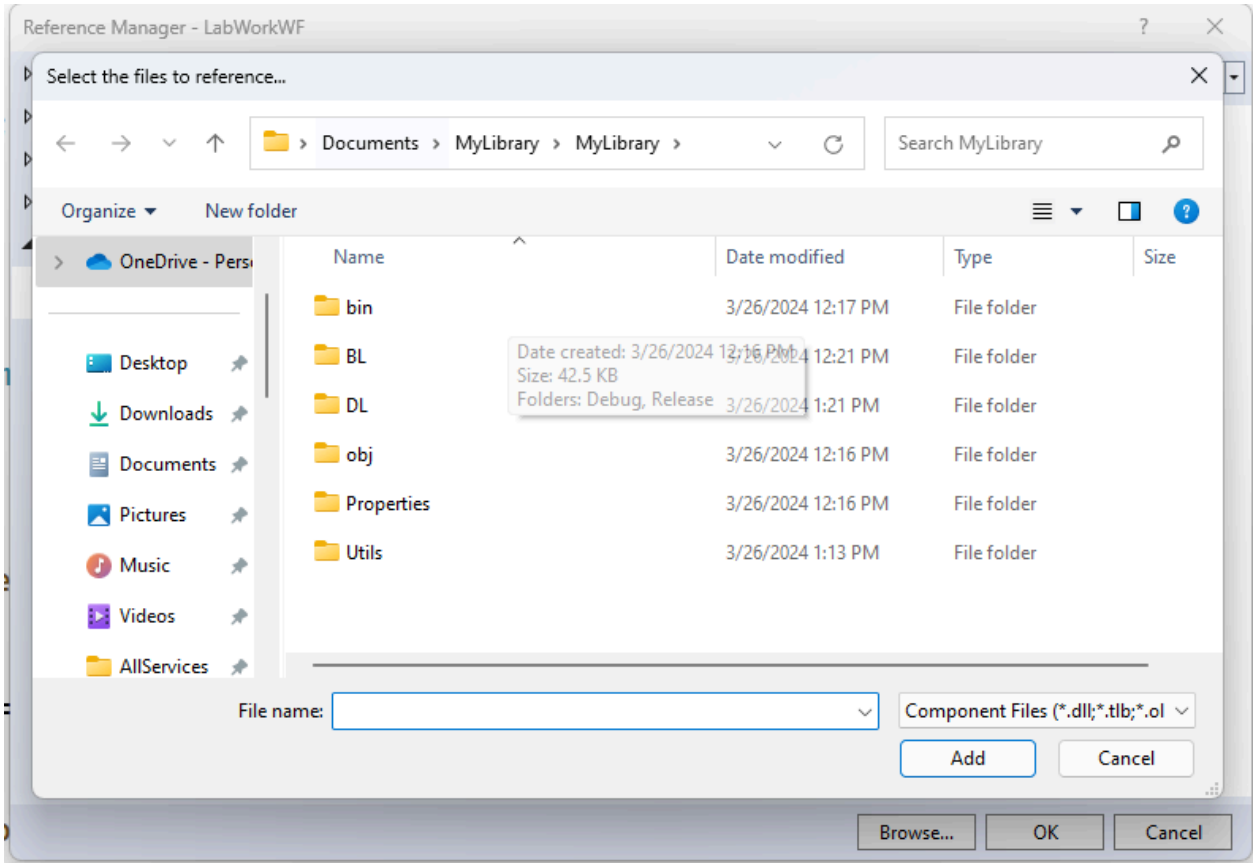
Click on Browse



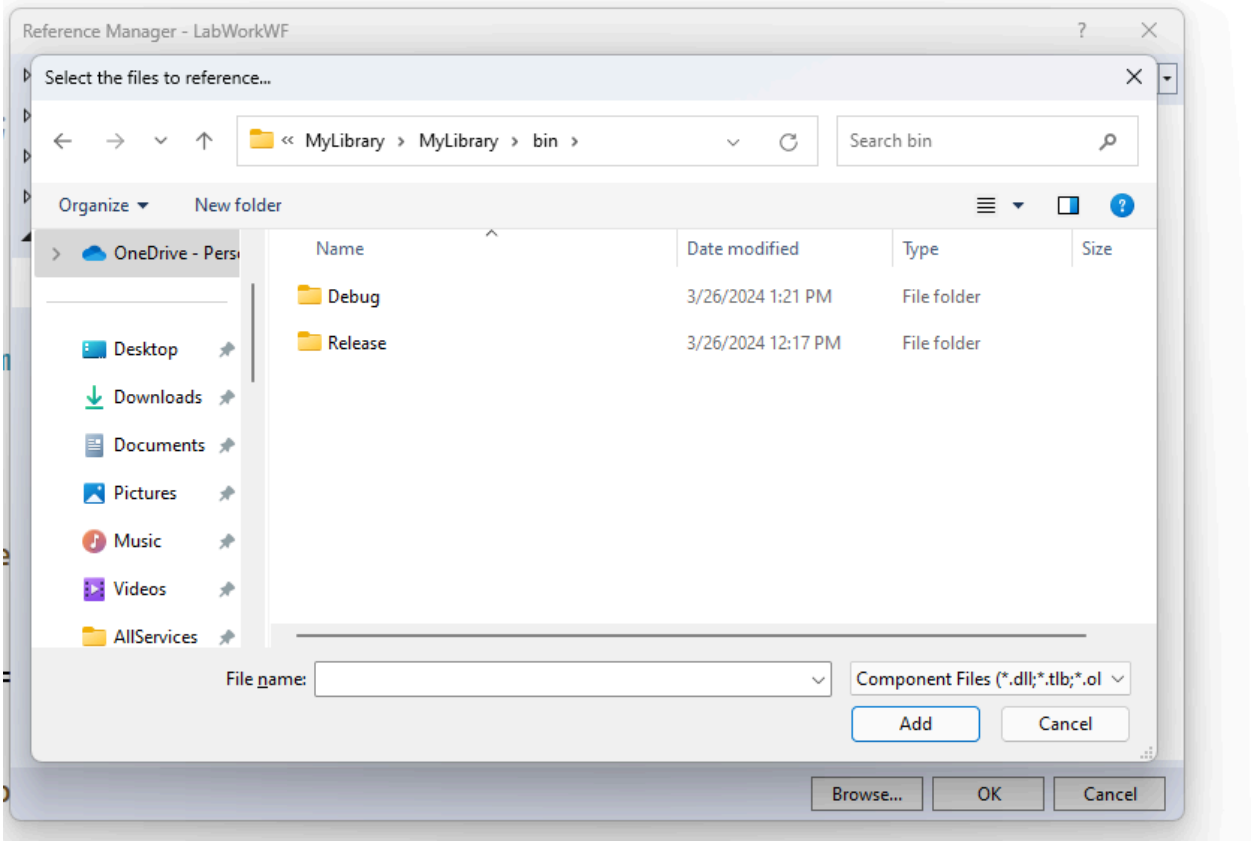
Now click on Browse button adjacent to Ok button at bottom



After browsing to the correct location, you will see your project in this window, like this:

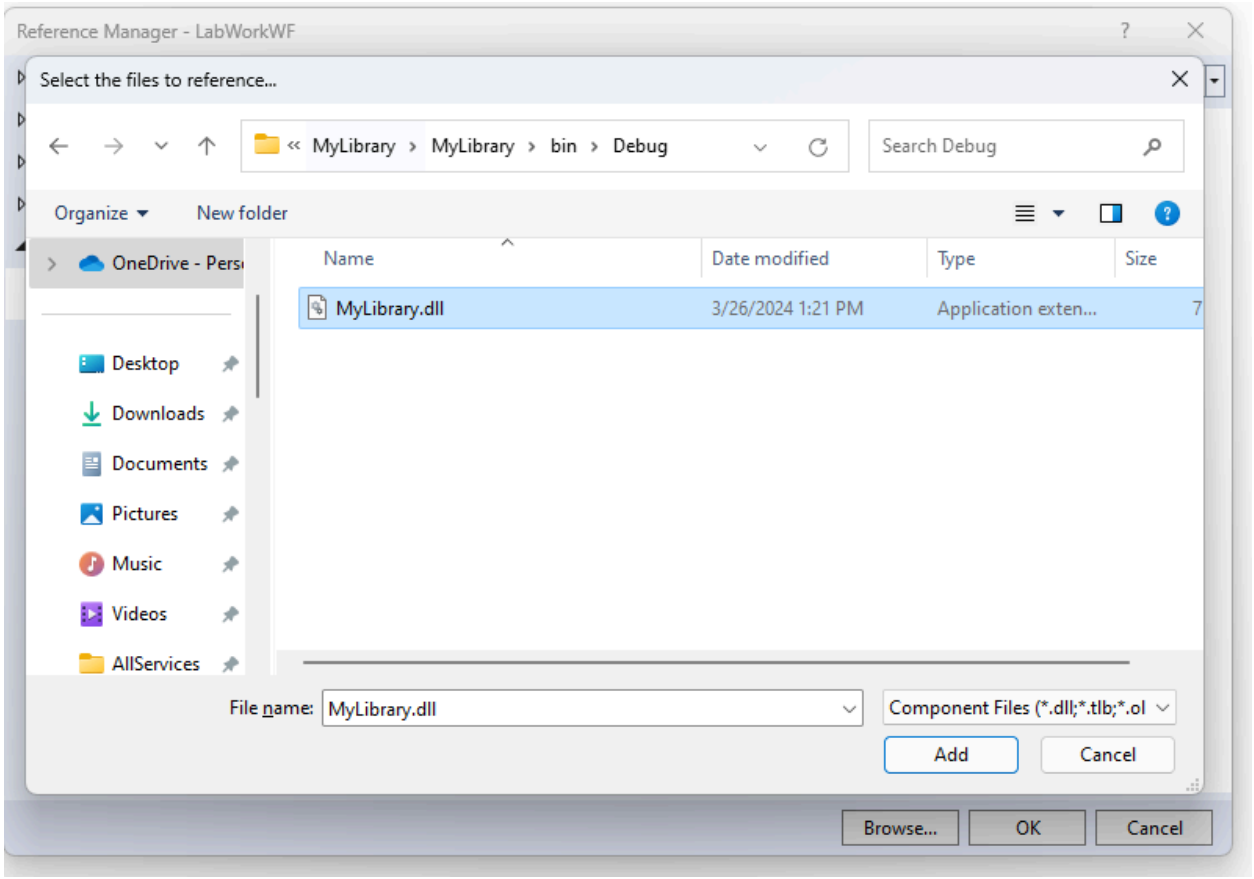


Now you can navigate into bin folder

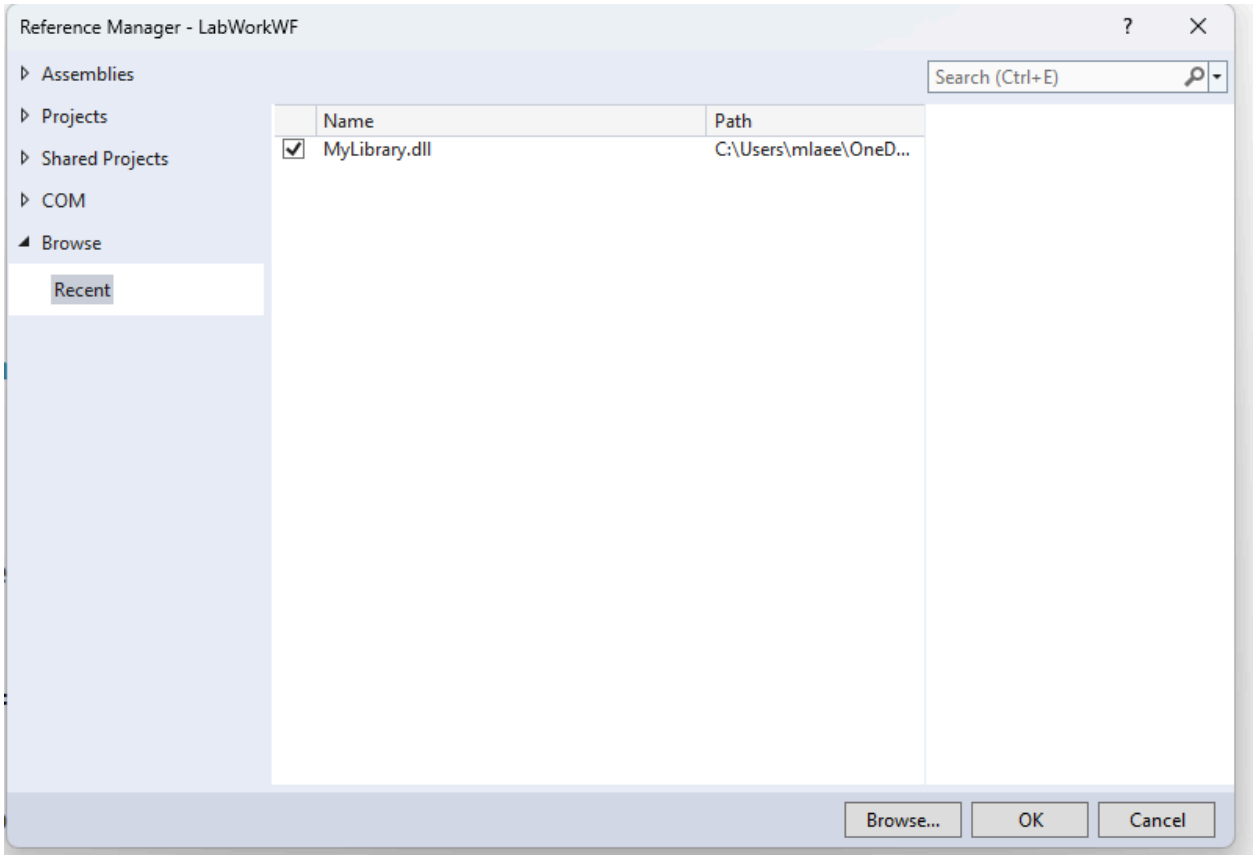


If you build your library project in debug mode, your library will be in the debug folder, and if you build your project in release mode, your library will be in release mode.

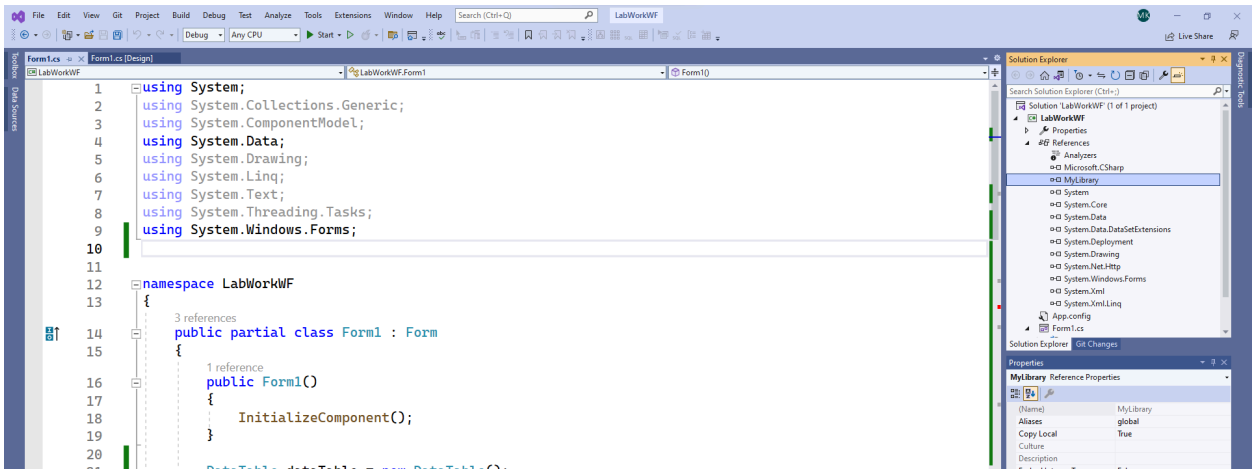
Select your library and click the 'Add' button


Reference Manager - LabWorkWF
Select the files to reference...
MyLibrary › MyLibrary › bin › Debug
Search Debug
Organize
New folder
OneDrive - Pers
Desktop
Downloads
Documents
Pictures
Music
Videos
AllServices
Name
Date modified
Type
Size
MyLibrary.dll
3/26/2024 1:21 PM
Application exten...
File name:
MyLibrary.dll
Component Files (*.dll;*.tlb;*.ol
Add
Cancel
Browse...
OK
Cancel

After clicking the 'Add' button, ensure that your library is selected with a checkbox in this window, then click the 'OK' button



After adding this, you can see your library in your **Solution Explorer.**



Now, let's begin calling your code from your library. Before writing the code, you can import your library namespaces into your code, or you can utilize Visual Studio's default feature to import them

```
using Labwork.BL;
using Labwork.DL;
```

Here is the code to load data from the StudentDL layer and add it to the DataGridView

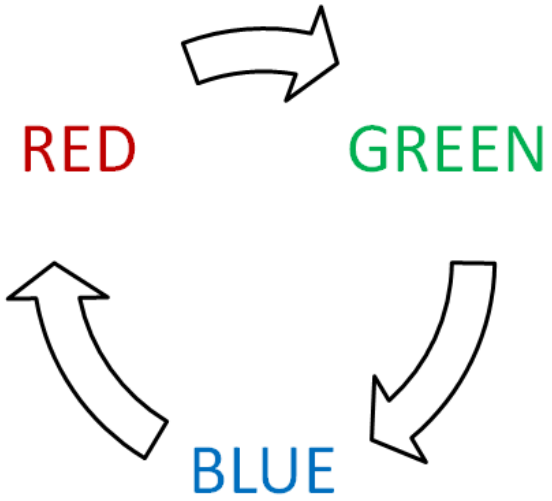
```
private void Form1_Load(object sender, EventArgs e)
{
    dataTable.Columns.Add("Roll", typeof(string));
    dataTable.Columns.Add("Name", typeof(string));
    dataTable.Columns.Add("Fsc", typeof(string));

    dataGridView1.DataSource = dataTable;

    

    //here the code to load data from DL Layer into your Datagridview on the form load
    List<Student> students = StudentDL.GetAllStudents();
    foreach(Student student in students)
    {
        dataTable.Rows.Add(student.Roll, student.Name, student.FSc);
    }
    dataGridView1.DataSource= dataTable;
}
```

Important Note:

When you make any changes to your library code, don't forget to rebuild the library; otherwise, you will not receive the changes in your project.

Challenge 01:

Create a graphical interface that shows a text box and two buttons with next and previous labels. When the user clicks on the next button it sets the textbox background color to the next color from the loop as given below and if the user presses the next button again it sets the background color of the text box with the next color. In case the user presses back it sets the previous color to the background of the textbox.



Good Luck and Best Wishes !!

Happy Coding ahead :)