I Wish I Knew How To …

Begin Programming Visual Studio 2019 C++/CLI and .NET on Windows Desktop

June 2019 Edition (1.0)

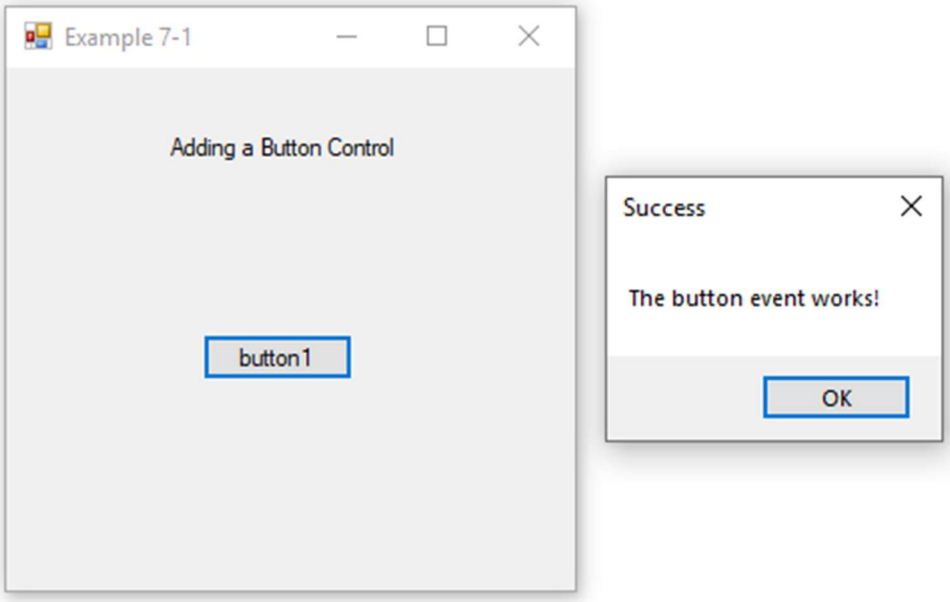
By Eugene Dakin

Table of Contents

Button

This example shows how to add a button that allows the execution of code when the button has been pressed by the user. The button control is very common is most programs.

Figure 67. Example 7-1: Button Control



This button control will create a new message box when Button1 has been pressed. Below are the steps to create this program and add the functionality to the button control.

To create this program, start a new CLR project:

4) File->New->Project
5) CLR Empty Project
6) Project Name: Example7-1 and select the location of the files

Configure the IDE by using the following steps:

4) Right-mouse-button-click on Example7-1 in the Solution Explorer and select Properties.
5) Select the following: Configuration Properties->Linker->System->Subsystem-> Windows (/SUBSYSTEM:WINDOWS) (*Select this*), Apply

6) Next, Type the word Main in the following section: Property-> Configuration Properties->Linker->Advanced->Entry Point->*Main.* Main is the standard starting method of the program.

7) Press the *Apply* button, and then press the *OK* button.

These steps now add a form to our Example7-1 project.

8) Select Example7-1 in Solution explorer and right-mouse click

9) Select Add->New Item->Visual C++->UI->Windows Form->Choose name of form (MyForm.h is good) and press the Add button. This may take a few seconds to complete.

10) You will get an error, and the next steps will prevent this from being an error.

11) In the solution Explorer, the MyForm.cpp code will need to be shown. Select Example7-1->Source Files->MyForm.cpp. There should only be one line of code.

Code 91. Example 7-1: MyForm.cpp

```
#include "MyButtonForm.h"
```

Replace the single line of code with the Edited code.

Code 92. Example 7-1: Edited MyForm.cpp

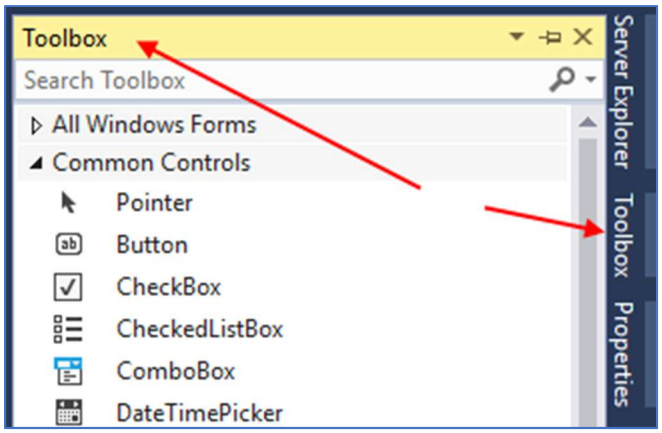
```
#include "MyForm.h"
using namespace System;
using namespace System::Windows::Forms;

[STAThreadAttribute]
void Main(array<String^>^ args) {
	Application::EnableVisualStyles();
	Application::SetCompatibleTextRenderingDefault(false);
	Example71::MyForm form; //Name of project, name of form, delete hyphen
	Application::Run(% form);
}
```

Just to make sure that everything is properly configured, press the *Local Windows Debugger* button to run the program and an empty form should appear.

12) The last step to show the form designer is to press File->Save All
13) Press File->Close Solution
14) Open the Example7-1 solution
15) Press the Window->Reset Window Layout button and press Yes.
16) Click on the Toolbox tab to show the Toolbox and the selection of controls.

Figure 68. Format Toolbox Tab and Toolbar



Click on the MyForm.h file in the Solution 'Example7-1' (1 project)->Example7-1->Header Files->MyForm.h and the new blank form is created. The next step is to add two controls, which are a button, and a label.

Double-click button1 to open an event, which executes code when the user presses the button. Add the following code to the button event.

Code 93. Example 7-1: button1 Event Code Snippet

```
private: System::Void Button1_Click(System::Object^ sender, System::EventArgs^ e) {
       MessageBox::Show("The button event works!", "Success", MessageBoxButtons::OK);
       }
```

When the button event is Button1_Click, this means that when the user presses on the mouse button which is a click-action, then execute the code in the curly brackets, which is to show a message box.

This example shows how to add a button to the program and make it show a message box when the user presses the button.

Index

The 'I Wish I Knew' series contains technical data and advice that makes sense and contains practical and numerous examples with explanations to allow you to ease into the steep programming curve.

This book shows how to start programming Windows C++/CLI Desktop applications. The C++/CLI language is slightly different than regular C++ because CLI is managed, which means that memory management is mostly managed by the IDE. The audience for this book is to introduce programmers to C++ with a language that minimizes memory management, and has significant speed improvements. Most programmers have learned programming with another language and wish to understand more of the C++ language, which this book is prepared for.

This book "I Wish I Knew How to … Begin Programming Visual Studio 2019 C++/CLI and .NET on Windows Desktop" shows you how to use the powerful C++/CLI language to make fast, native programs.

This book provides the steps to manually create each example, as old legacy code that is stored on various online systems are outdated and do not convert properly or run. Being able to create each example from scratch makes this book a valuable part of your computer programming library. Where possible, visual controls are added to the forms, and much programming is performed visually to make it easier for programming, maintenance, and future compatibility.

There are 10 chapters and contains over 300 pages with over 55 example programs, and the code for each example accompanies this book.

This is one of many books at xDevLibrary. This book can be purchased at http://XojoLibrary.com where many great resources are available.

Happy programming!

Eugene

Eugene Dakin MBA, Ph.D., P.Chem., is an author of Xojo reference materials and has many years of experience in the programming industry. Another great reference book is *I Wish I Knew How To … Implement API Declares on Windows*.

ISBN: 978-1-927924-25-9