HNDIT1012 Visual Application Programming

Week 1

M.A.C Jiffriya B.Sc, M.Sc, M.Phil(pera)

Visual Application Programming

- Course Code HNDIT 1012
- No of Credits: 4
- Lecture   : 2 hours
- Tutorial / Practical : 4 hours
- Student Activity : 7 hours

Course Aims

- Visual programming languages are widely used for the rapid development of graphical applications. This subject will introduce the visual application programming environment and to create simple event driven GUI based applications using C#

Learning Outcomes (LO)

- LO1: Use GUI based integrated environment in effective coding, compiling, and debugging programs.
- LO2: Analyse program requirements
- LO3: Develop program modules to handle common, simple programming problems.
- LO4: Use strings, files, and streams in programs.
- LO5: Design user interfaces with GUI widgets and perform event handling
- LO6: Design/develop programs with GUI interfaces

What is a program?

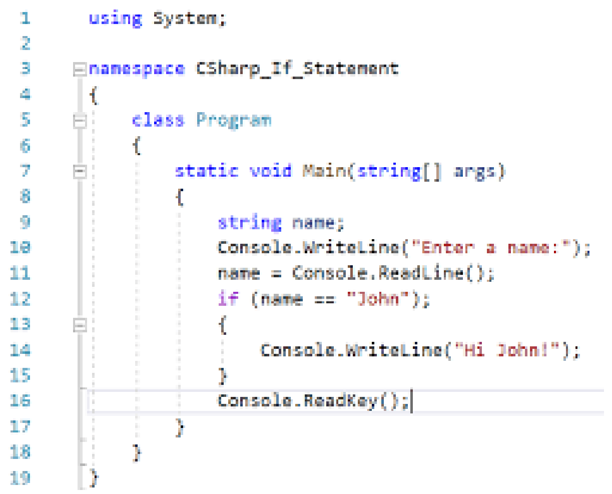
- A computer program is a sequence or set of instructions in a programming language for a computer to execute.
- Computer programs are one component of software.

Programming languages

- A computer programming language is a language used to write computer programs, which involves a computer performing some kind of computations.
- Thousands of different programming languages have been created, and more are being created every year.
- Many programming languages are written in an imperative form (i.e., as a sequence of operations to perform) while other languages use the declarative form (i.e. the desired result is specified, not how to achieve it).
- Eg: C++, C#, VisualBasic, Java etc..

Source code

- A computer program in its human-readable form is called source code.
- Source code needs another computer program to execute because computers can only execute their native machine instructions.
- Therefore, source code may be translated to machine instructions using the language's translators (compiler / Interpreter)



```
using System;

namespace CSharp_If_Statement
{
    class Program
    {
        static void Main(string[] args)
        {
            string name;
            Console.WriteLine("Enter a name:");
            name = Console.ReadLine();
            if (name == "John");
            {
                Console.WriteLine("Hi John!");
            }
            Console.ReadKey();
        }
    }
}
```

Executable file

- An EXE file is an executable program you can run in Microsoft Windows. Most EXE files contain either Windows applications or application installers.

Language Translators

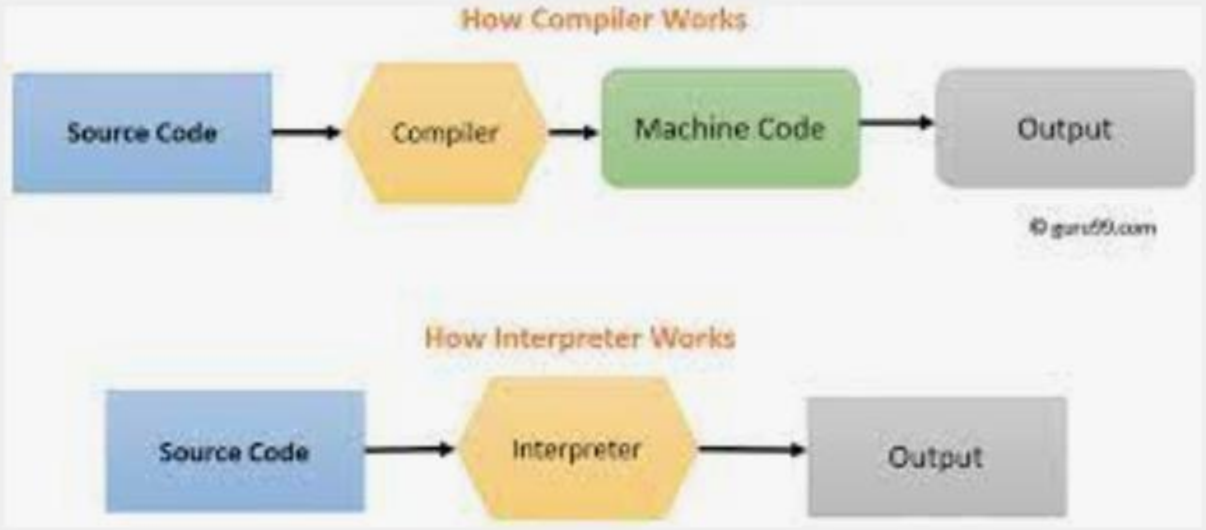
- Language translators allow computer programmers to write sets of instructions in specific programming languages. These instructions are converted by the language translator into machine code. The computer system then reads these machine code instructions and executes them.
  - Compiler
  - Interpreter

Compiler

- Once all errors have been corrected translates the entire source code as a single unit.
- After the compilation process the source program is no more required
- Faster execution

Interpreter

- Converts source code statements into equivalent machine language statements and execute.
- Source program is required for every execution
- Slow execution


How Compiler Works
Source Code
Compiler
Machine Code
Output
© guru99.com
How Interpreter Works
Source Code
Interpreter
Output

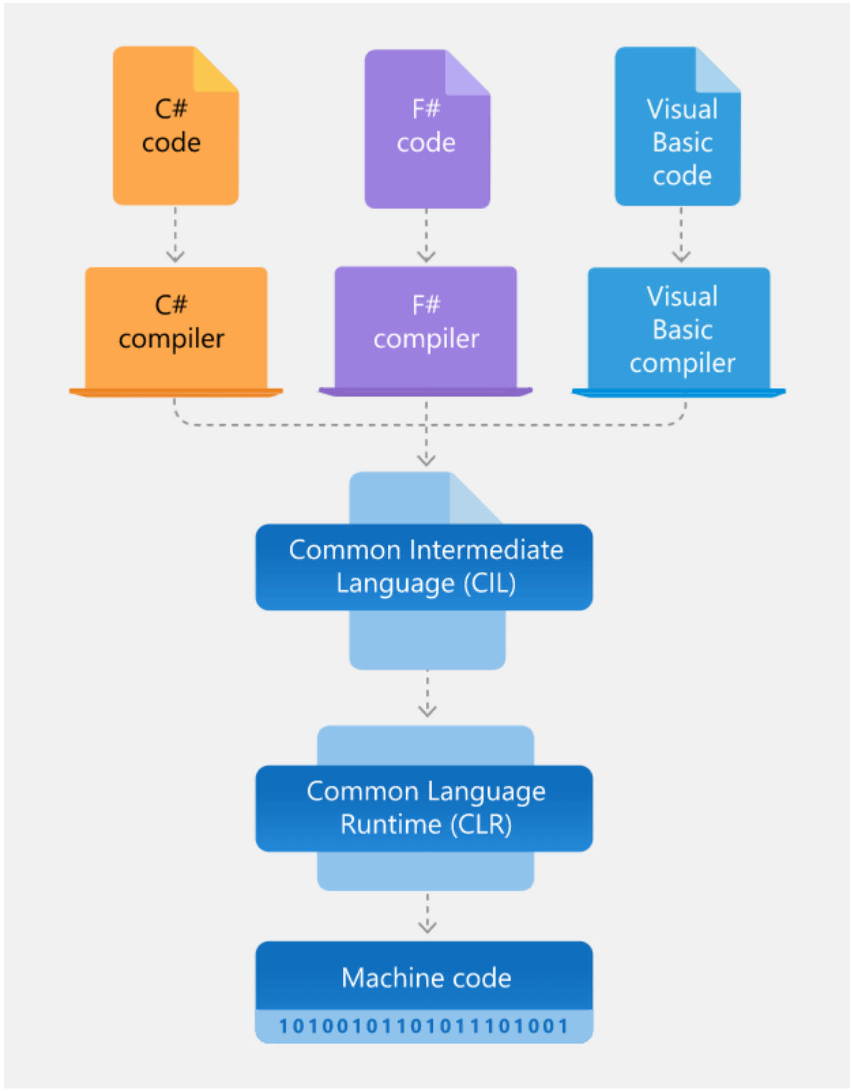
NET Framework

- NET Framework is a software development framework for building and running applications on Windows. . NET Framework is part of the . NET platform, a collection of technologies for building apps for Linux, macOS, Windows, iOS, Android, and more.

Architecture of .NET Framework ..

- .NET applications are written in the C#, F#, or Visual Basic programming language. Code is compiled into a language-agnostic Common Intermediate Language (CIL).
- Compiled code is stored in assemblies—files with a .dll or .exe file extension.
- When an app runs, the CLR takes the assembly and uses a just-in-time compiler (JIT) to turn it into machine code that can execute on the specific architecture of the computer it is running on.

Architecture of .NET Framework

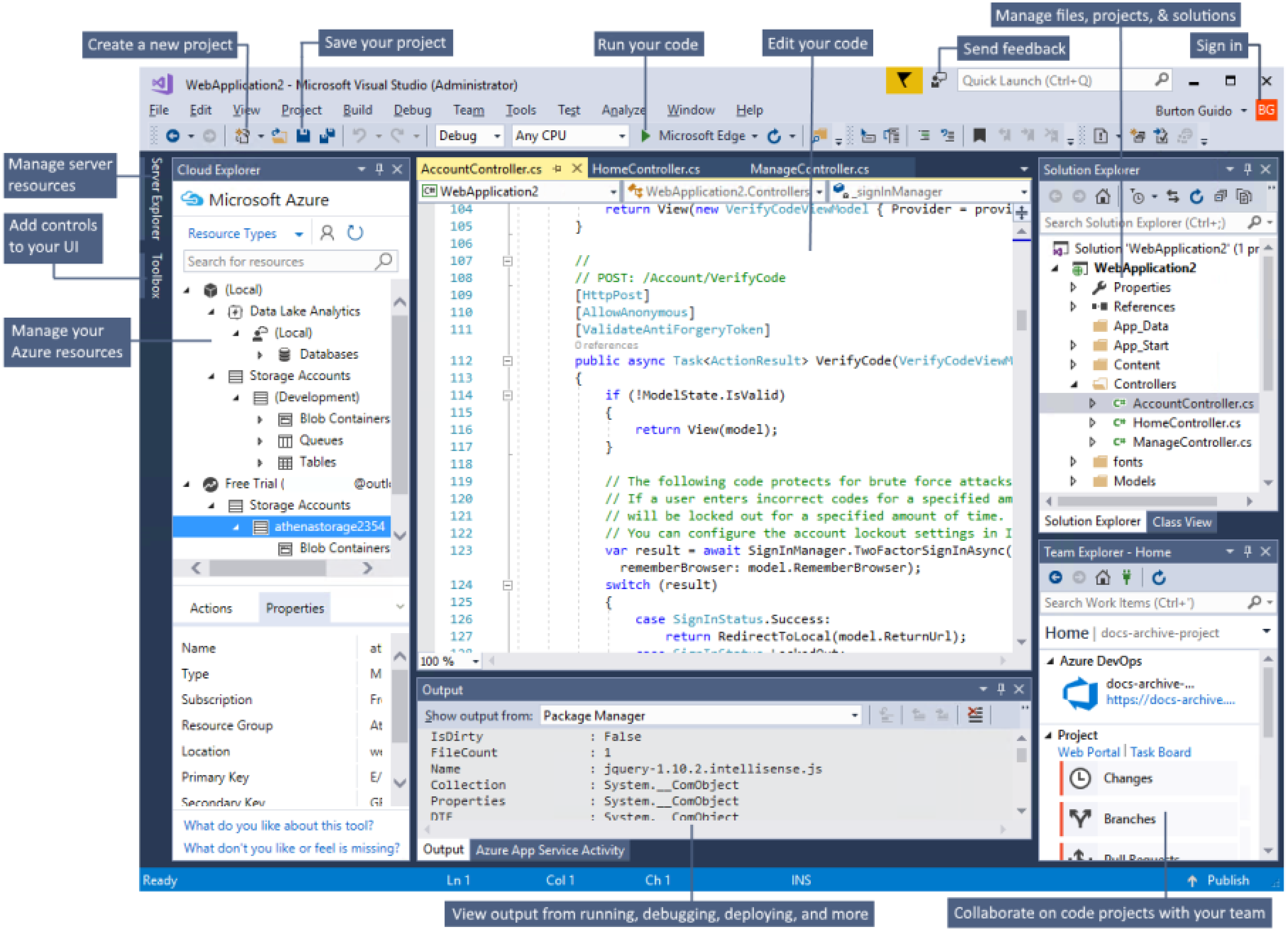


Introduction to IDE (Visual studio 2022)

- An integrated development environment (IDE) is software for building applications that combines common developer tools into a single GUI.
- Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.

Visual studio 2022..

- The Visual Studio integrated development environment is a creative launching pad that you can use to edit, debug, and build code, and then publish an app.
- An integrated development environment (IDE) is a feature-rich program that can be used for many aspects of software development.
- Visual Studio includes compilers, code completion tools, graphical designers, and many more features to ease the software development process.

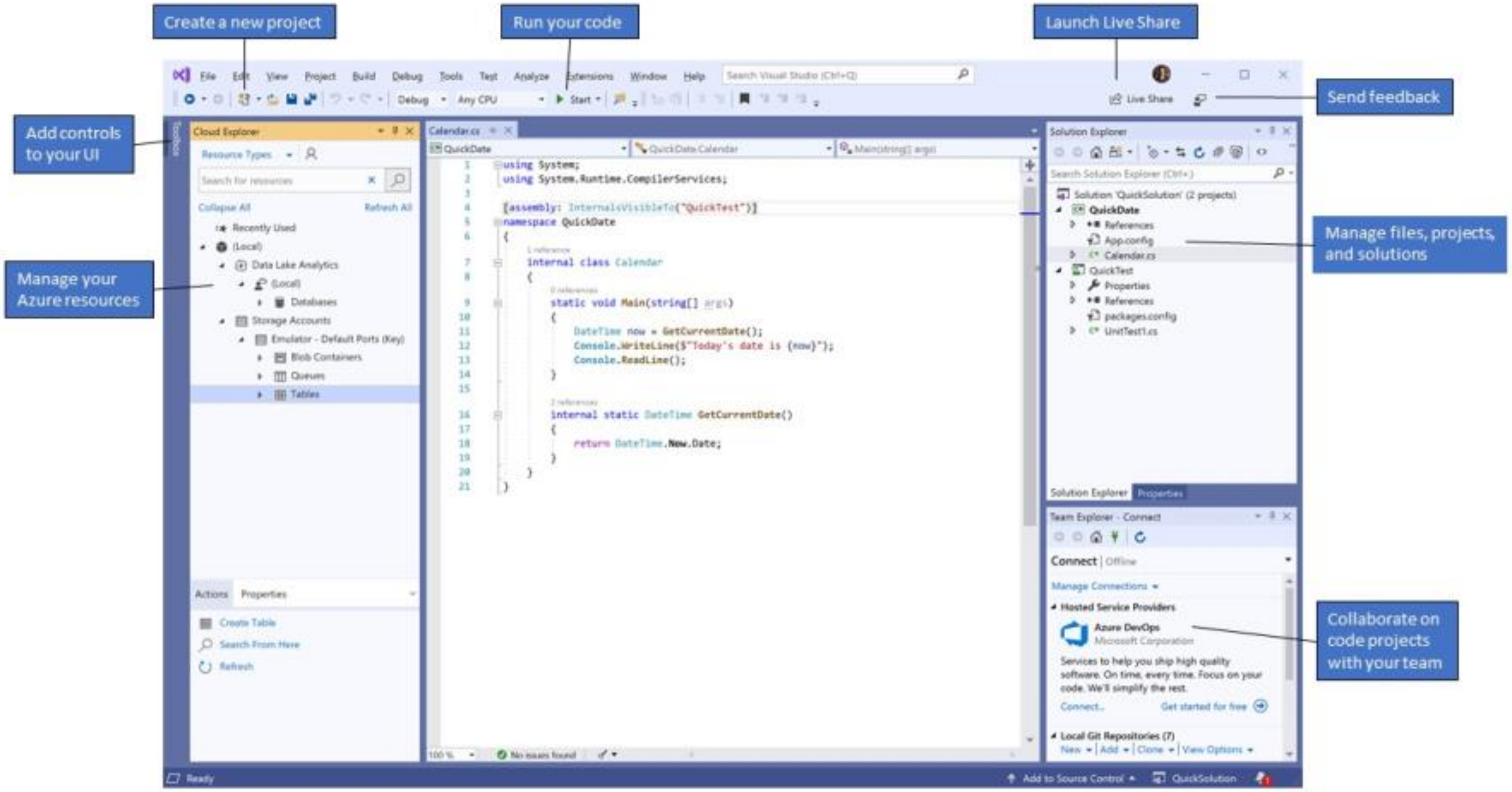


```
        return View(new VerifyCodeViewModel { Provider = provi
    }

    //
    // POST: /Account/VerifyCode
    [HttpPost]
    [AllowAnonymous]
    [ValidateAntiForgeryToken]
    0 references
    public async Task<ActionResult> VerifyCode(VerifyCodeViewM
    {
        if (!ModelState.IsValid)
        {
            return View(model);
        }

        // The following code protects for brute force attacks
        // If a user enters incorrect codes for a specified am
        // will be locked out for a specified amount of time.
        // You can configure the account lockout settings in I
        var result = await SignInManager.TwoFactorSignInAsync(
          rememberBrowser: model.RememberBrowser);
        switch (result)
        {
            case SignInStatus.Success:
                return RedirectToLocal(model.ReturnUrl);
```

```
IsDirty                  : False
FileCount                : 1
Name                     : jquery-1.10.2.intellisense.js
Collection               : System.__ComObject
Properties               : System.__ComObject
DTE                      : System.__ComObject
```


Create a new project
Run your code
Launch Live Share
Send feedback
Add controls to your UI
Manage your Azure resources
Manage files, projects, and solutions
Collaborate on code projects with your team

```
using System;
using System.Runtime.CompilerServices;

[assembly: InternalsVisibleTo("QuickTest")]
namespace QuickDate
{
    internal class Calendar
    {
        static void Main(string[] args)
        {
            DateTime now = GetCurrentDate();
            Console.WriteLine($"Today's date is {now}");
            Console.ReadLine();
        }

        internal static DateTime GetCurrentDate()
        {
            return DateTime.Now.Date;
        }
    }
}
```

Visual studio 2022..

- Solution Explorer (top right) lets you view, navigate, and manage your code files. Solution Explorer can help organize your code by grouping the files into solutions and projects.
- The editor window (center), where you'll likely spend a majority of your time, displays file contents. This is where you can edit code or design a user interface such as a window with buttons and text boxes.
- The Output window (bottom center) is where Visual Studio sends notifications such as debugging and error messages, compiler warnings, publishing status messages, and more. Each message source has its own tab.

Installing VisualStudio 2022

Before you begin installing Visual Studio:

- Check the system requirements. These requirements help you know whether your computer supports Visual Studio 2022.
- Apply the latest Windows updates. These updates ensure that your computer has both the latest security updates and the required system components for Visual Studio.
- Reboot. The reboot ensures that any pending installs or updates don't hinder your Visual Studio install.
- Free up space. Remove unneeded files and applications from your system drive by, for example, running the Disk Cleanup app.

System Requirements for VisualStudio 2022 - Hardware

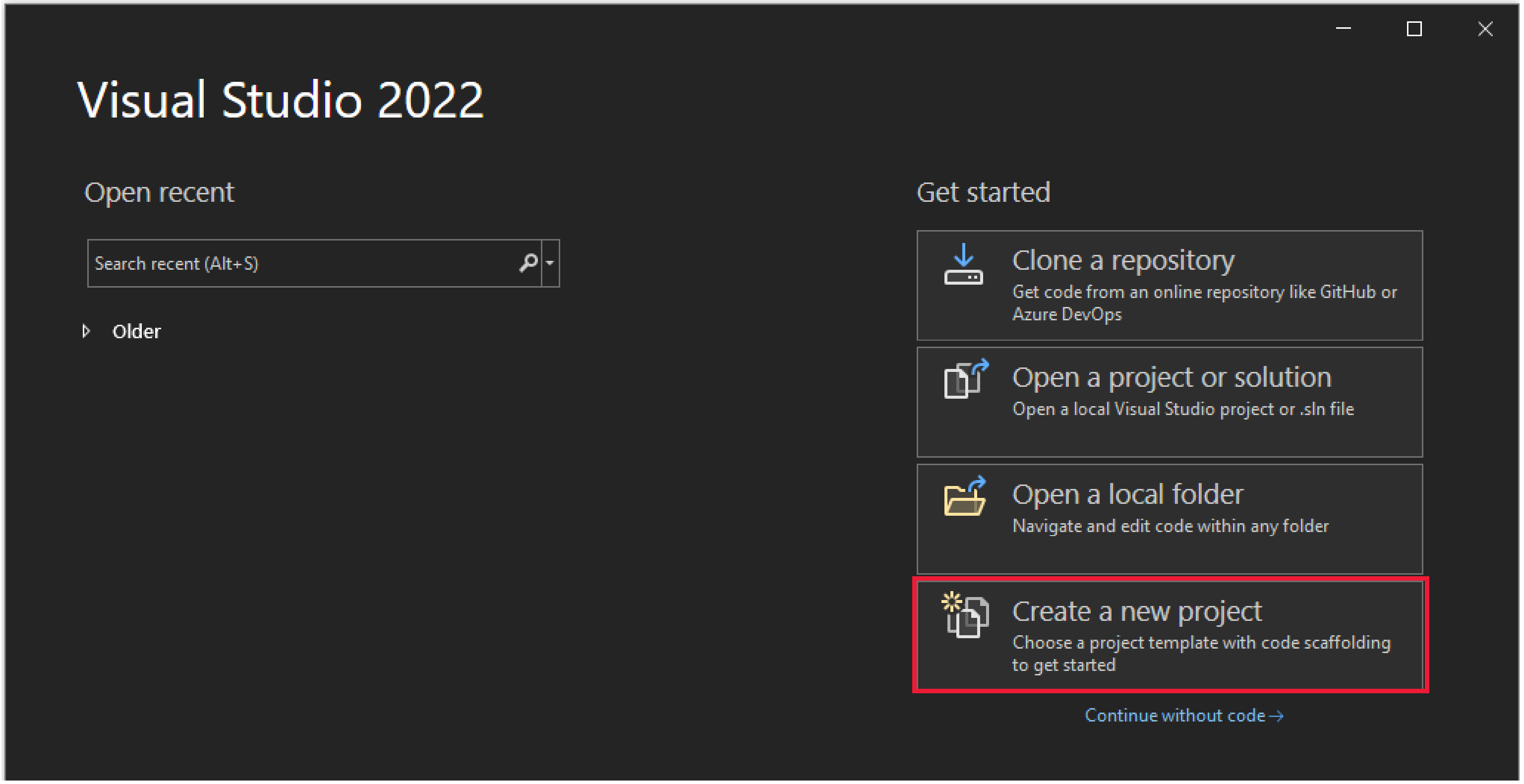
- 1.8 GHz or faster 64-bit processor; Quad-core or better recommended. ARM processors are not supported.
- Minimum of 4 GB of RAM. Many factors impact resources used; we recommend 16 GB RAM for typical professional solutions.
- Windows 365: Minimum 2 vCPU and 8 GB RAM. 4 vCPU and 16 GB of RAM recommended.
- Hard disk space: Minimum of 850 MB up to 210 GB of available space, depending on features installed; typical installations require 20-50 GB of free space.
- We recommend installing Windows and Visual Studio on a solid-state drive (SSD) to increase performance.
- Video card that supports a minimum display resolution of WXGA (1366 by 768); Visual Studio will work best at a resolution of 1920 by 1080 or higher.

Download Visual Studio

- Download link
  https://visualstudio.microsoft.com/downloads
- Follow the instructions in the following link.
  https://docs.microsoft.com/en-us/visualstudio/install/install-visual-studio?view=vs-2022

Creating a window based project.

- Open Visual Studio.
- On the start window, choose Create a new project.



Visual Studio 2022

Open recent

Search recent (Alt+S)

Older

Get started

Clone a repository
Get code from an online repository like GitHub or Azure DevOps

Open a project or solution
Open a local Visual Studio project or .sln file

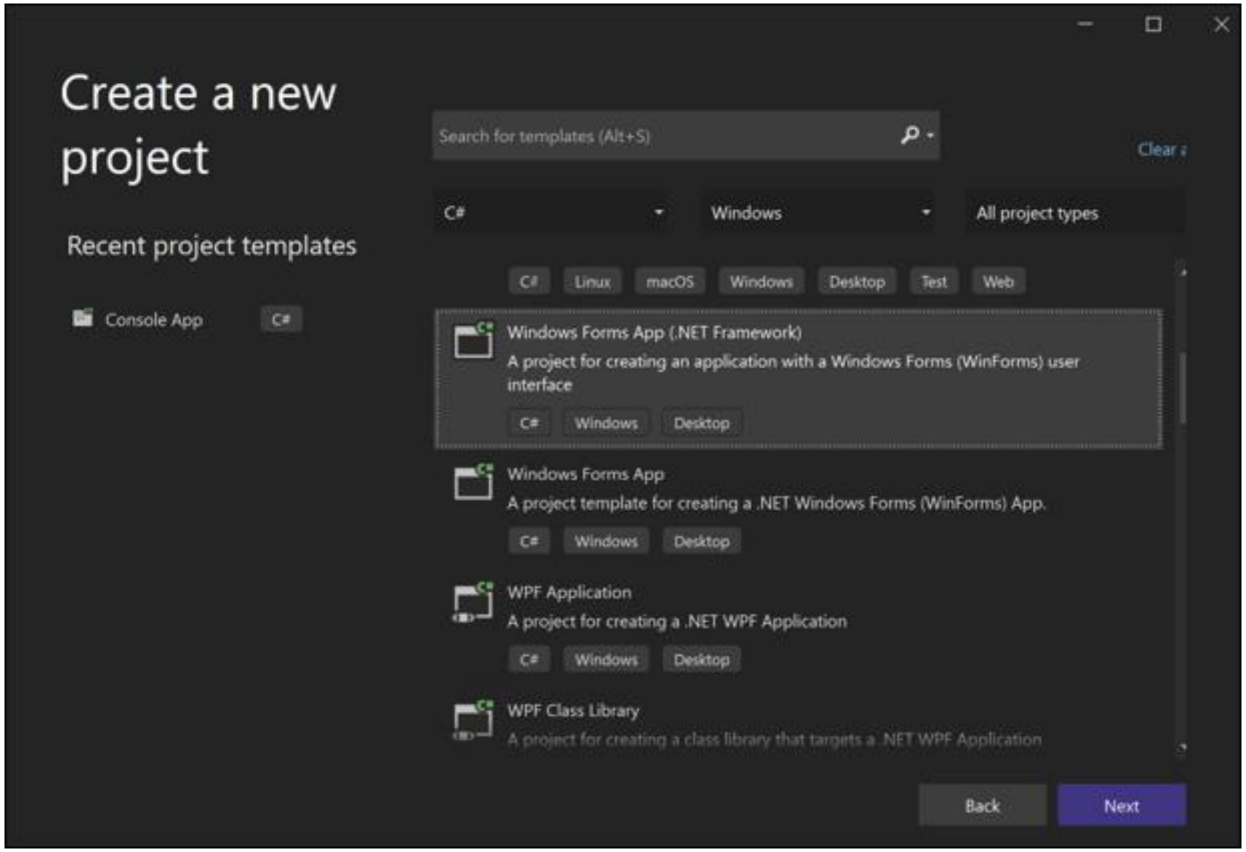
Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

Continue without code →

Creating a window based project…

- On the Create a new project window, choose the Windows Forms App (. NET Framework) template for C#. ...



Create a new project

Recent project templates

Console App C#

Search for templates (Alt+S)

C# | Windows | All project types

C# Linux macOS Windows Desktop Test Web

Windows Forms App (.NET Framework)
A project for creating an application with a Windows Forms (WinForms) user interface
C# Windows Desktop

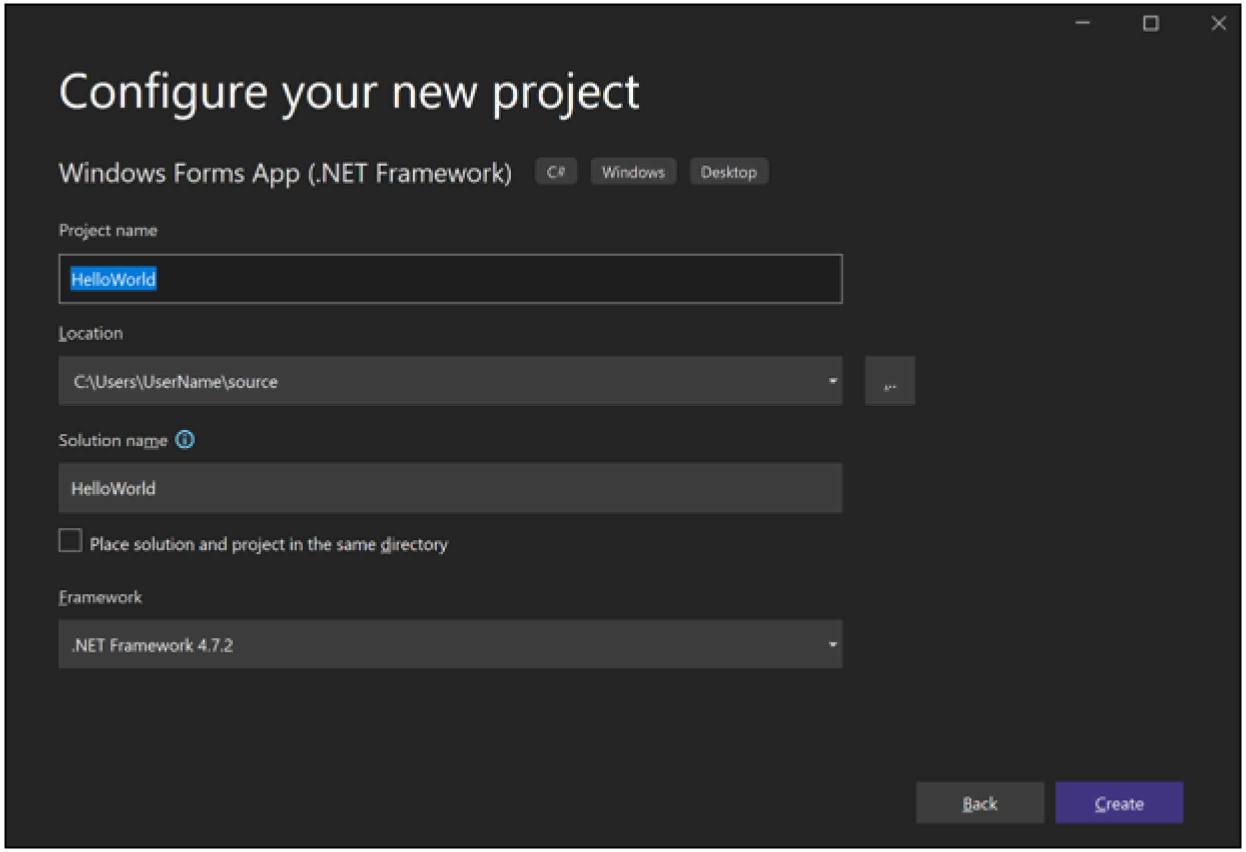
Windows Forms App
A project template for creating a .NET Windows Forms (WinForms) App.
C# Windows Desktop

WPF Application
A project for creating a .NET WPF Application
C# Windows Desktop

WPF Class Library
A project for creating a class library that targets a .NET WPF Application

Creating a window based project…

- In the Configure your new project window, type or enter HelloWorld in the Project name box. Then, choose Create.



Configure your new project

Windows Forms App (.NET Framework)

Project name: HelloWorld

Location: C:\Users\UserName\source

Solution name: HelloWorld

☐ Place solution and project in the same directory

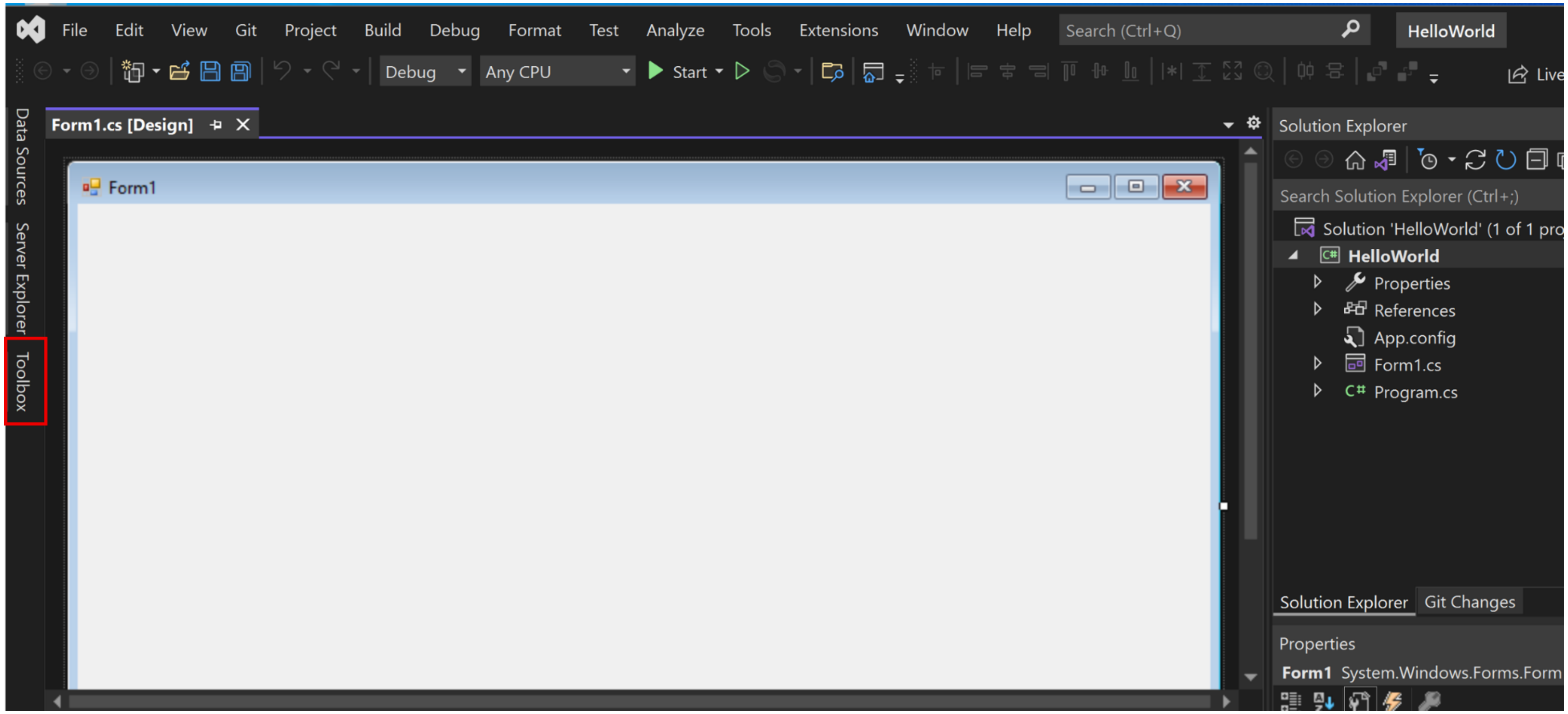
Framework: .NET Framework 4.7.2

Create the application

- After you select your C# project template and name your file, Visual Studio opens a form for you. A form is a Windows user interface. We'll create a "Hello World" application by adding controls to the form, and then we'll run the app.

Create the application..

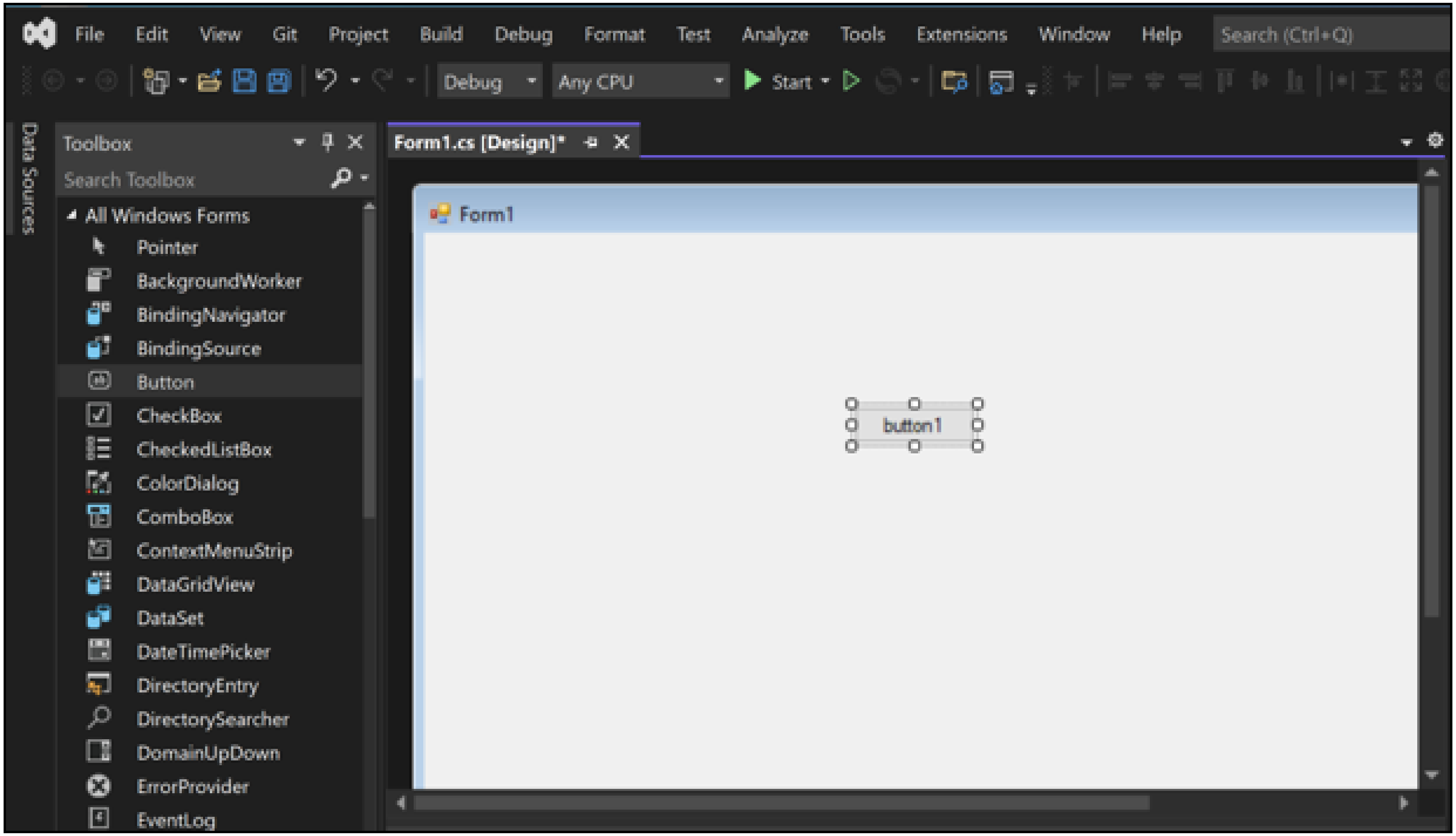
- Select Toolbox to open the Toolbox fly-out window.



Create the application..

- Select the Pin icon to dock the Toolbox window.
- Select the Button control and then drag it onto the form.
- In the Properties window, locate Text, change the name from button1 to Click this, and then press Enter.
- In the Design section of the Properties window, change the name from button1 to btnClickThis, and then press Enter.

Create the application..



Create the application..

- Select the Label control from the Toolbox window, and then drag it onto the form and drop it beneath the Click this button.

- In either the Design section or the (DataBindings) section of the Properties window, change the name of label1 to lblHelloWorld, and then press Enter.

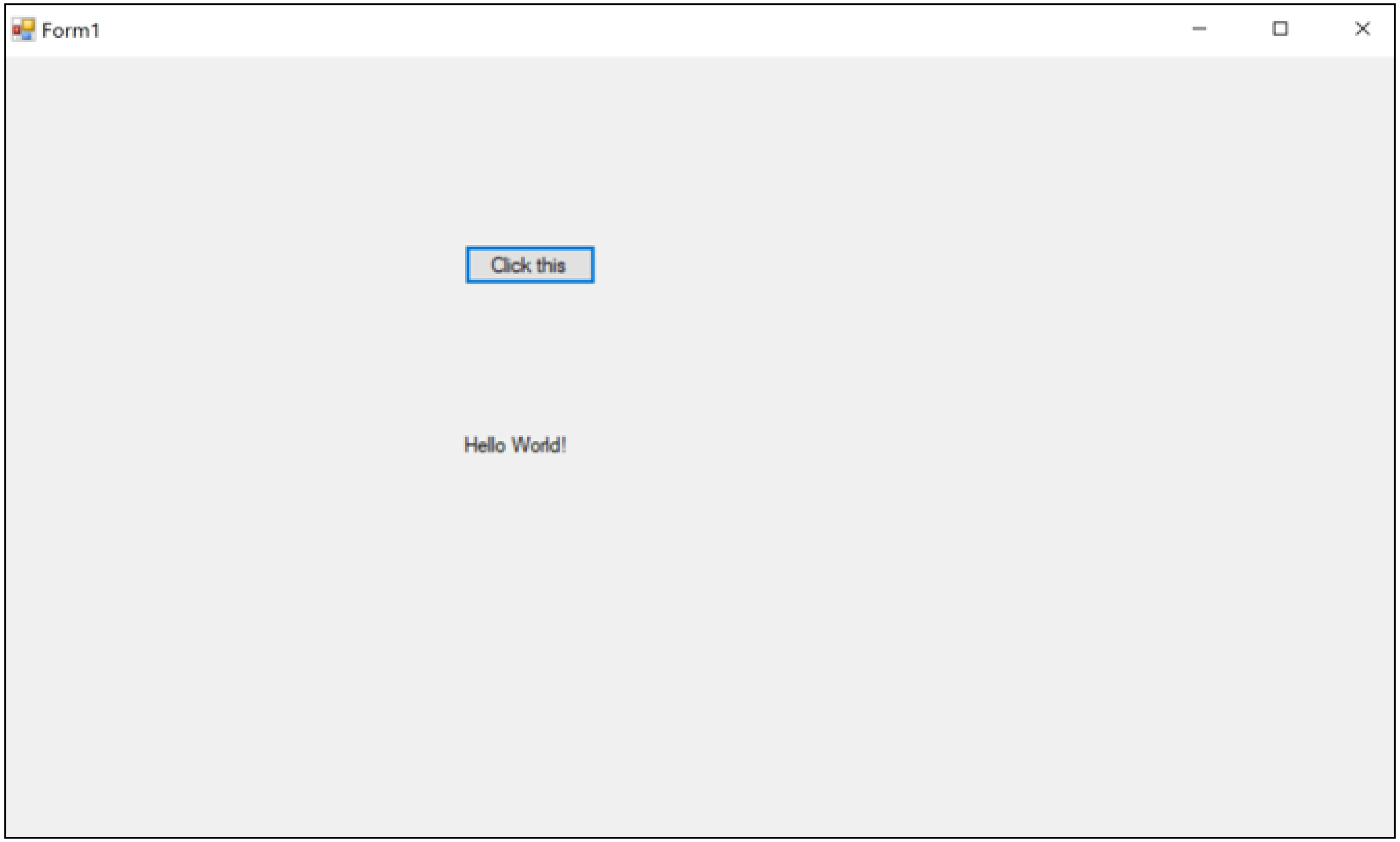
Add code to the form

- In the Form1.cs [Design] window, double-click the Click this button to open the Form1.cs window.
- (Alternatively, you can expand Form1.cs in Solution Explorer, and then choose Form1.)
- In the Form1.cs window, after the private void line, type or enter lblHelloWorld.Text = "Hello World!";

Run the application

- Select the Start button to run the application.
- Several things will happen. In the Visual Studio IDE, the Diagnostics Tools window will open, and an Output window will open, too. But outside of the IDE, a Form1 dialog box appears. It will include your Click this button and text that says label1.
- Select the Click this button in the Form1 dialog box. Notice that the label1 text changes to Hello World!.

Run the application…



Hello World!

Close the Form1 dialog box to stop running the app.

Thank You