What is C#:

C# is an object-oriented programming language that enables developers to build a variety of secure and robust application that run on .NET

Why Use C#

- It's one of the most popular programming languages well versed in the private sector
- Simple to learn and use
- C# is an object-oriented language which gives a clear structure to programs and allows code to be resused.
- When you learn C#, you can easily switch over Java, C++

Applications written in C# use the .NET Framework

What is .NET Framework?

.NET Framework: It's a software development framework for building and running applications on Windows.

The architecture of .NET Framework are the Common Language Runtime and .NET Framework class library.

Common Language Runtime (CLR) is the execution engine that handles running applications. It' provides services like thread management, garbage collection, type-safety, exception handling, and more.

Class Library provides a set of API's and types of common functionality. It provides types for strings, dates, numbers, etc. The class library includes APIs for reading and writing files, connecting to databases, drawing and more.

.NET applications are written in C#, the code is complied into language Common Intermediate Language (CIL). Compiled code is stored in assemblies – files with a .dll or .exe file extension.

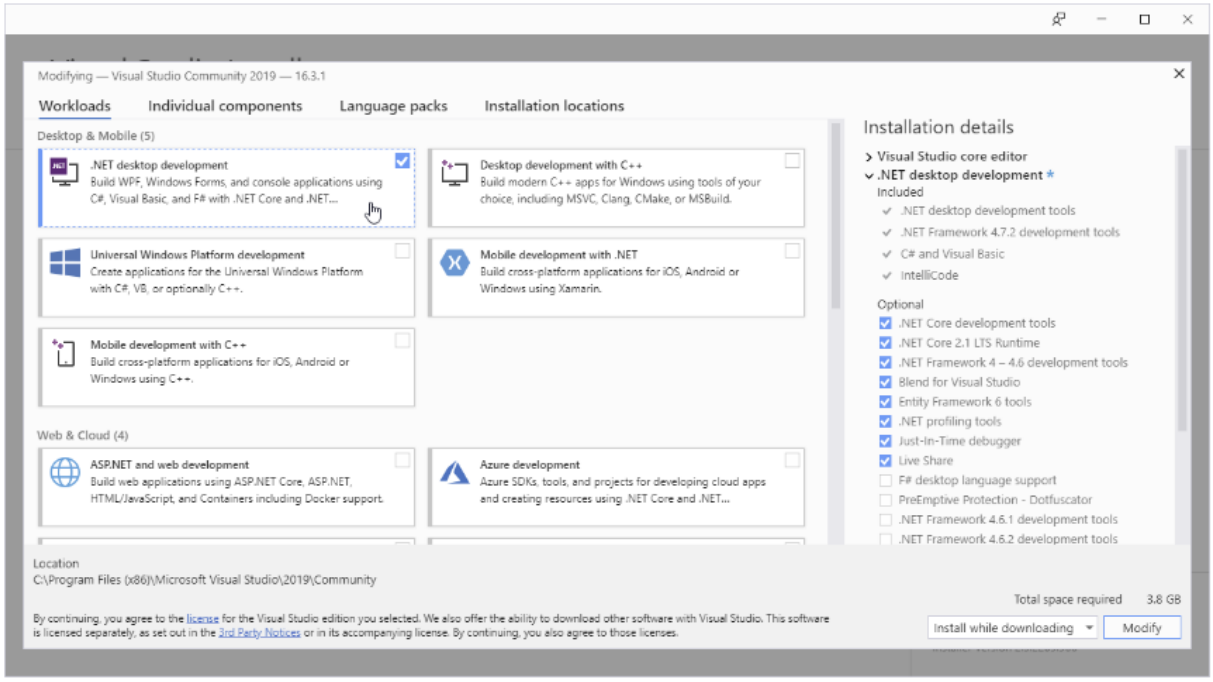
When the app runs, the CLR takes the assembly and uses a just in time compiler (JIT) to turn it into machine code that can execute on the specific architecture of the computer it is running on.

The newest version of .NET Framework 5.0

The most stable and supported version is .NET Framework 4.8

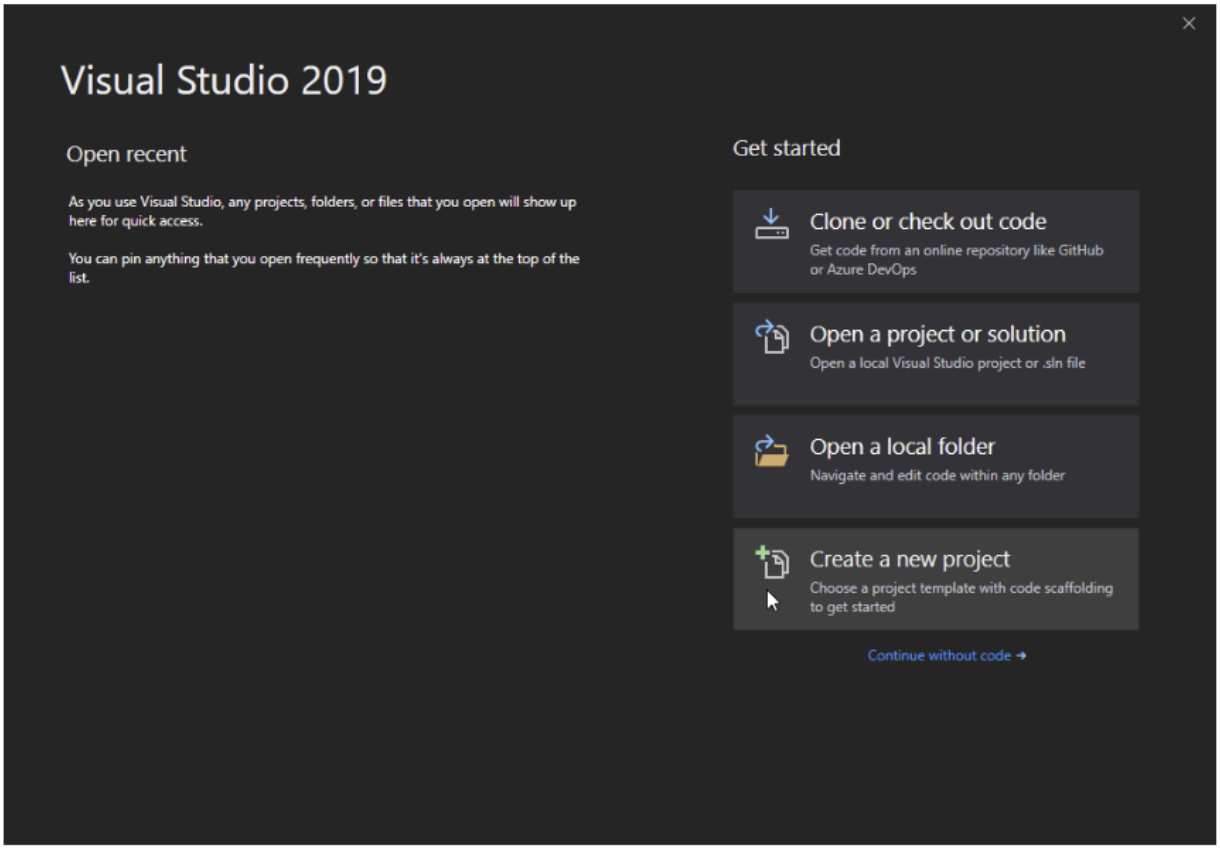
How to install Visual Studio 2019 and set up your project solution for .NET environment:

When the Visual Studio Installer is downloaded and installed, choose the .NET workload, and click on Modify/Install button:

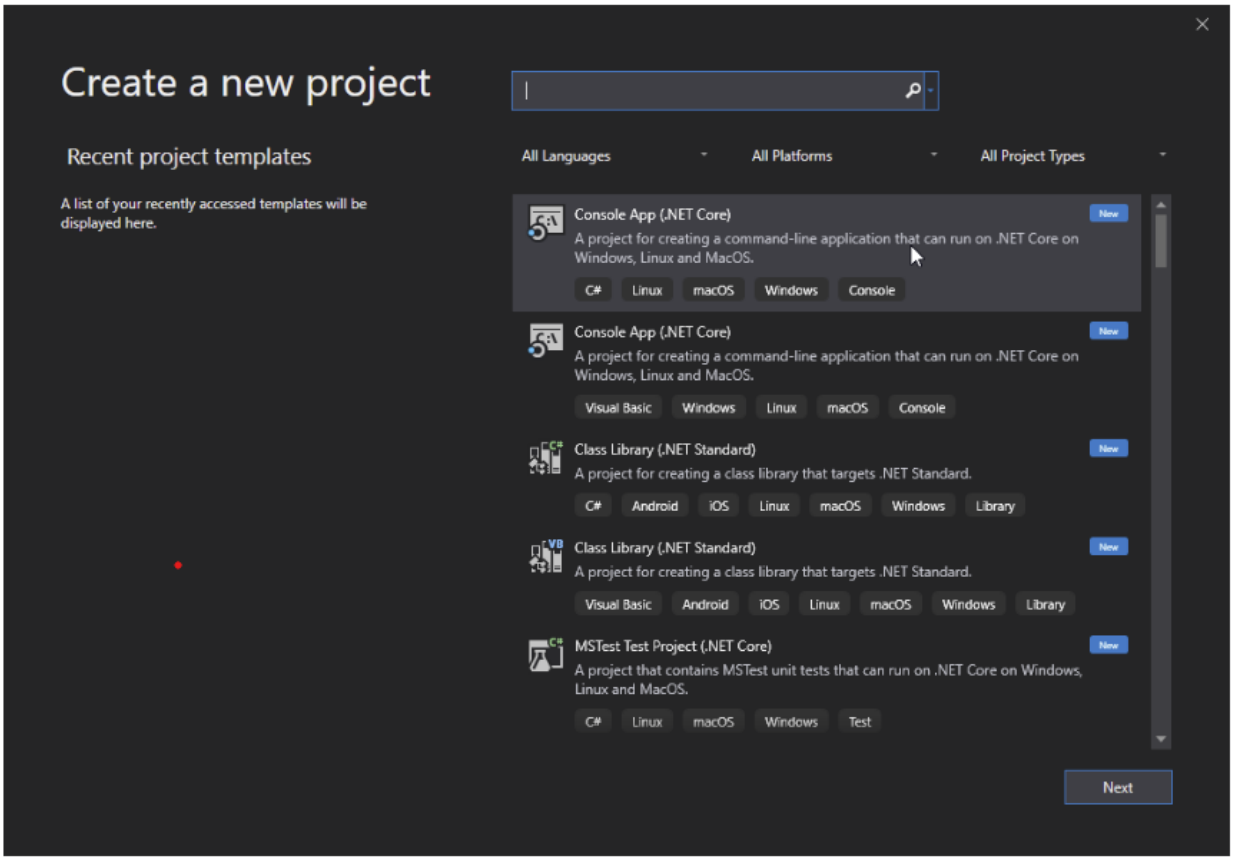


After the installation is complete, click on the **Launch** button to get started with Visual Studio.

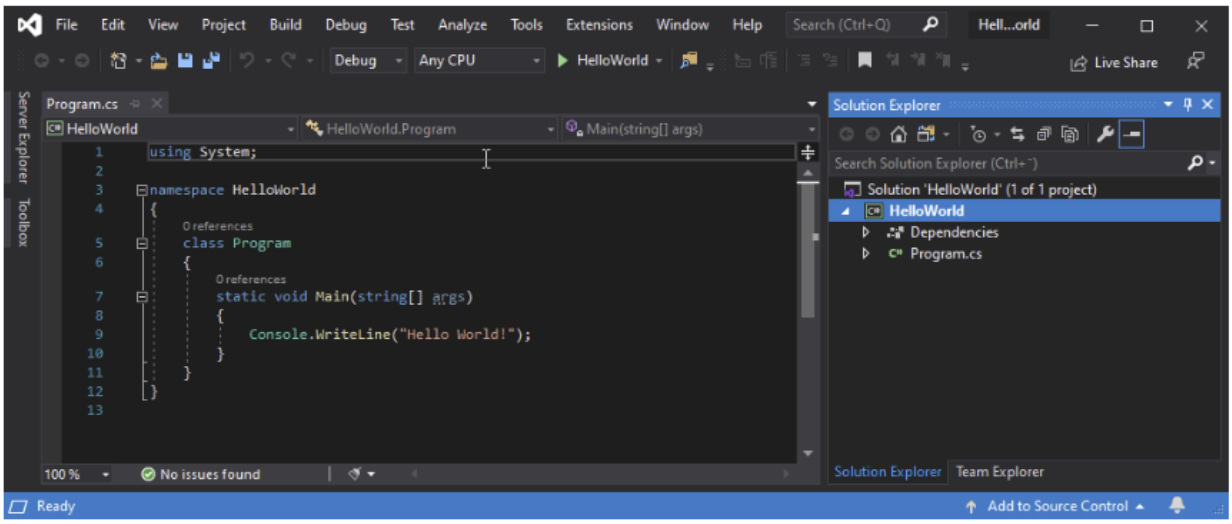
On the start window, choose CREATE A NEW PROJECT :



Then choose “Console App (.NET Core)” from the list and click on the Next button:



Enter the name for your project, and click on the Create button. Your code for your project should look like this:



Run the program by pressing the F5 button on your keyboard or click on “DEBUG” -> “Start Debugging”. This will compile and execute your code.

CONGRULATIONS, you have written and executed your first C# program!

Let’s talk about C# Syntax

Let’s look a Program.cs file.

Line 1: **using System** means that we can use classes from the System namespace.

Line 3: **namespace** is used to organize your code, and it is a container for classes and other namespaces.

Line 4: The **curly braces {} marks** the beginning and the end of a block of code.

Line 5: class is a container for data and methods, which brings functionality to your program. Every line of code that runs in C# must be inside a class.

Line 7: **The Main method,** any code inside its curly brackets {} will be executed.

Line 9: **Console** is a class of the System namespace, which has a WriteLine() method that is used to output/print text.

*If you didn’t use System line, you would have to write

System.Console.WriteLine() to print/output text.

Every C# statement ends with a semicolon ;

- C# is **CASE-SENSITIVE**: “MyClass” and “myclass” have DIFFERENT MEANING

The following lines will appear in your program it’s the standard line of codes that will be used in every program file.

Top Key Concepts to Learn with C# Language:

- Learn C# Comments which and be used to explain C# code and to make readable.
- Variables (int, double, char, string, bool, constants).
- Data Types
- Type Casting (is when you assigned a value of one data type to another type)
- Type Conversion Methods
- User Input
- Operators
- Strings
- Boolean
- If- Else
- Switch
- While Loop
- For Loop
- Arrays
- Methods
- Method Parameters (information can be passed to methods as parameter. Parameters act as variables inside the method.
- Method Overloading (multiple methods can have the same name with different parameters.
- Classes/ Objects
- Exceptions
- Files
- C# OOP

Focus on learning these concepts and learn how to implement them into your introductory software application.