

OOP LAB 1 (22 Sept 22)

Introduction to C#

By: Miss Tia



- ☐ Basics of C# and Unity
- ☐ Submit lab work, where? via telegram (pm me)
- ☐ Submit when? You have 1 week to complete (before next OOP lab)
- ☐ Feel free to ask and give feedbacks
- ☐ First 6 weeks C#, last 2 weeks will be on Unity

- C# (pronounced "See Sharp") is a modern, object-oriented, and type-safe programming language.
- Has roots in the C family of languages and will be immediately familiar to C, C++, Java, and JavaScript programmers.
- Games made with C# include temple run
- C# default language for unity

Download Unity + Visual Studio

Download Unity

Download Unity

Welcome! You're here because you want to download Unity, the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences.

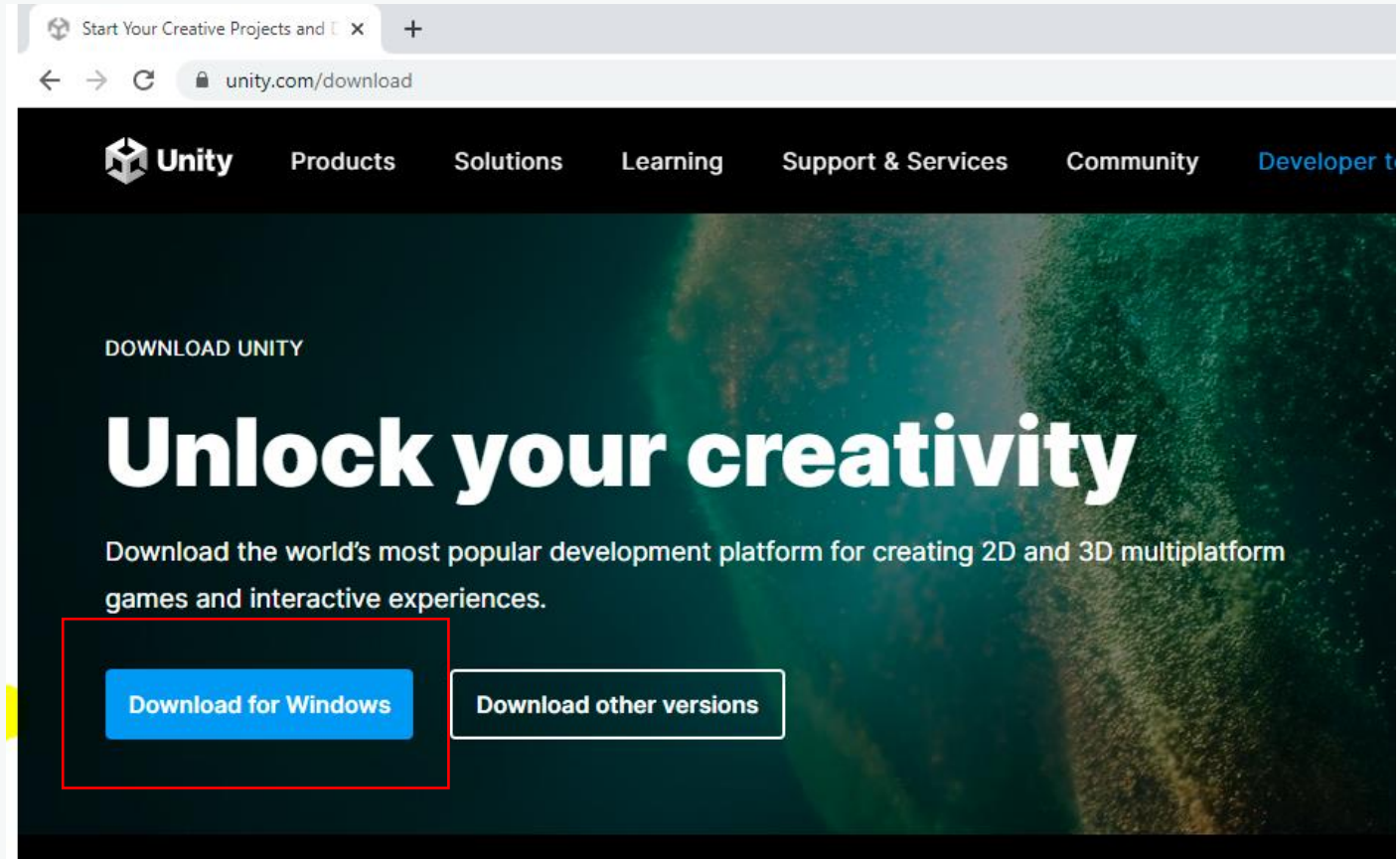
Before you download choose the version of Unity that's right for you.

Choose your Unity + download

Download Unity Hub



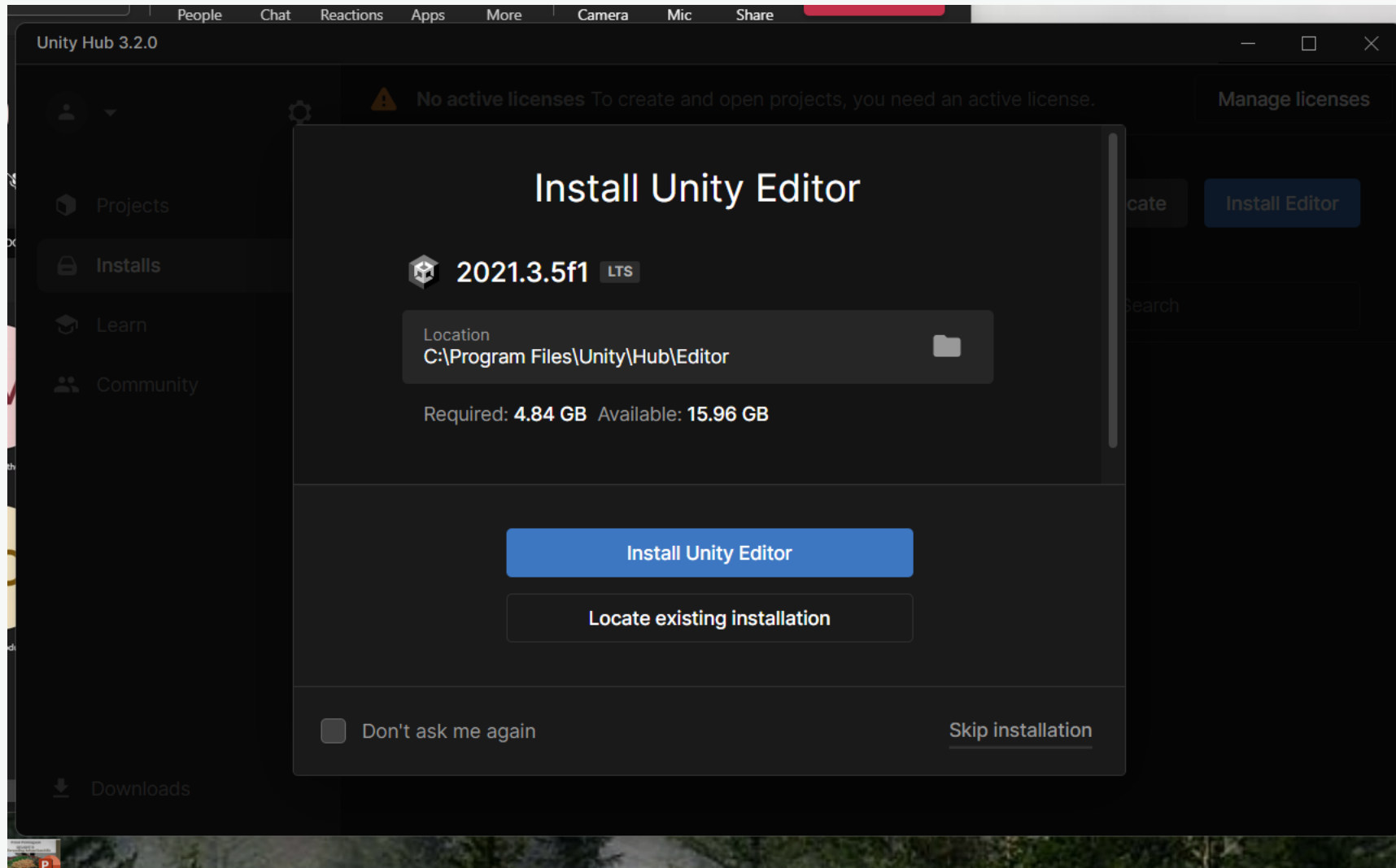
[Learn more about the new Unity Hub here.](#)



The screenshot shows the Unity website's download page. The browser address bar displays 'unity.com/download'. The page features a dark header with the Unity logo and navigation links: Products, Solutions, Learning, Support & Services, Community, and Developer t. The main content area has a dark background with a green, abstract, textured pattern on the right. The text 'DOWNLOAD UNITY' is in small caps. Below it, the headline 'Unlock your creativity' is in large white font. A sub-headline reads: 'Download the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences.' At the bottom, there are two buttons: 'Download for Windows' (highlighted with a red box) and 'Download other versions' (outlined with a white box).

Download Unity + Visual Studio

It will show this



Configuration Visual Studio

Tick the ones in the box, this will download unity and visual studio together in 1 go

Install Unity 2021.3.5f1 LTS

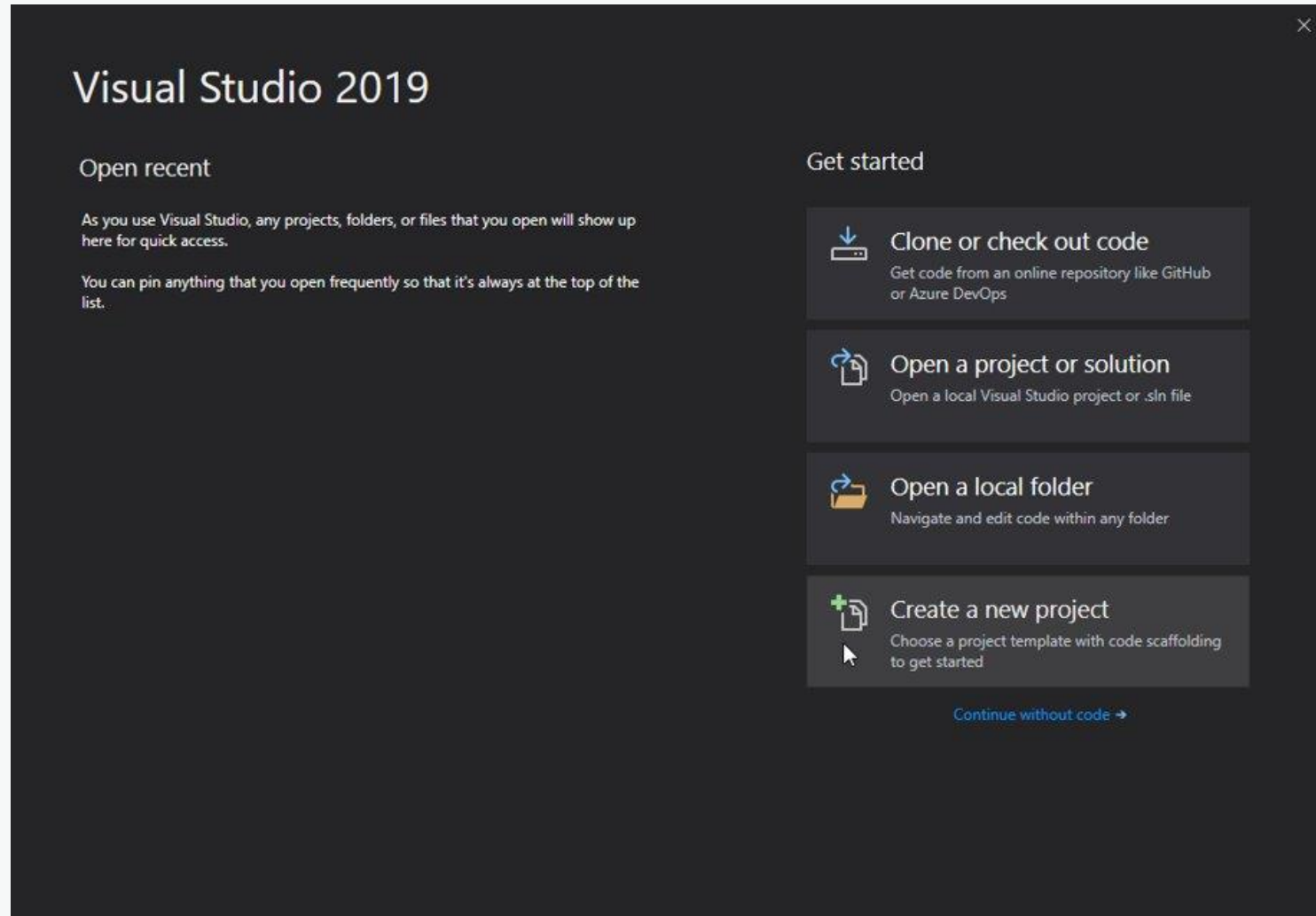
Add modules Required: **5.46 GB** Available: **284.37 GB**

	DOWNLOAD SIZE	SIZE ON DISK
DEV TOOLS		
<input checked="" type="checkbox"/> Microsoft Visual Studio Community 2019	Installed	1.24 GB
PLATFORMS		
<input checked="" type="checkbox"/> Android Build Support	364.38 MB	1.86 GB
<input checked="" type="checkbox"/> OpenJDK	145.91 MB	67.2 MB
<input checked="" type="checkbox"/> Android SDK & NDK Tools	1.27 GB	4.21 GB
<input type="checkbox"/> iOS Build Support	405.99 MB	1.78 GB
<input type="checkbox"/> tvOS Build Support	401.79 MB	1.76 GB
<input type="checkbox"/> Linux Build Support (IL2CPP)	53.42 MB	221.86 MB
<input type="checkbox"/> Linux Build Support (Mono)	53.43 MB	221.32 MB

Back **Install**

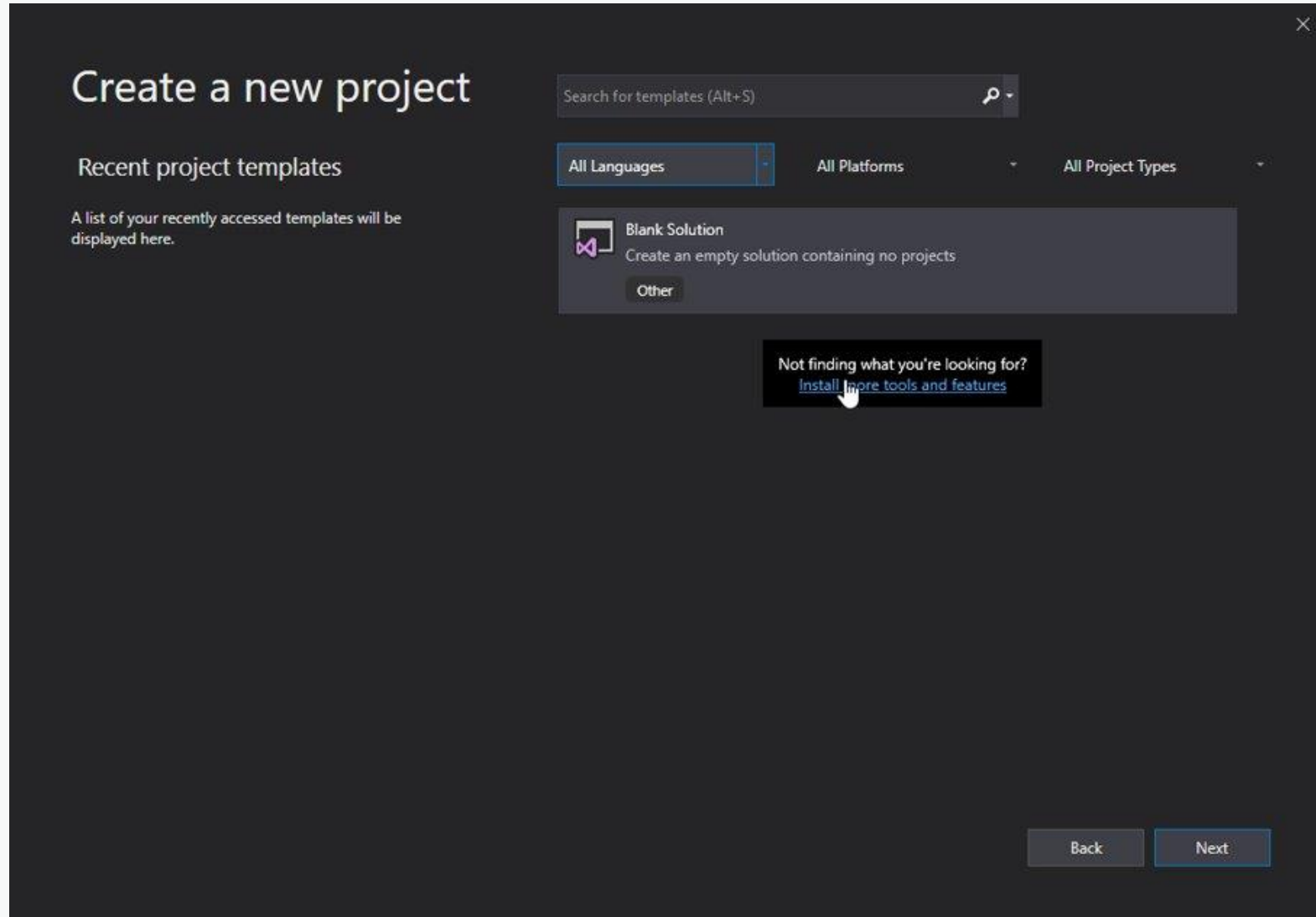
Configuration Visual Studio

click on the **Launch** button to get started with Visual Studio.
On the start window, choose **Create a new project**



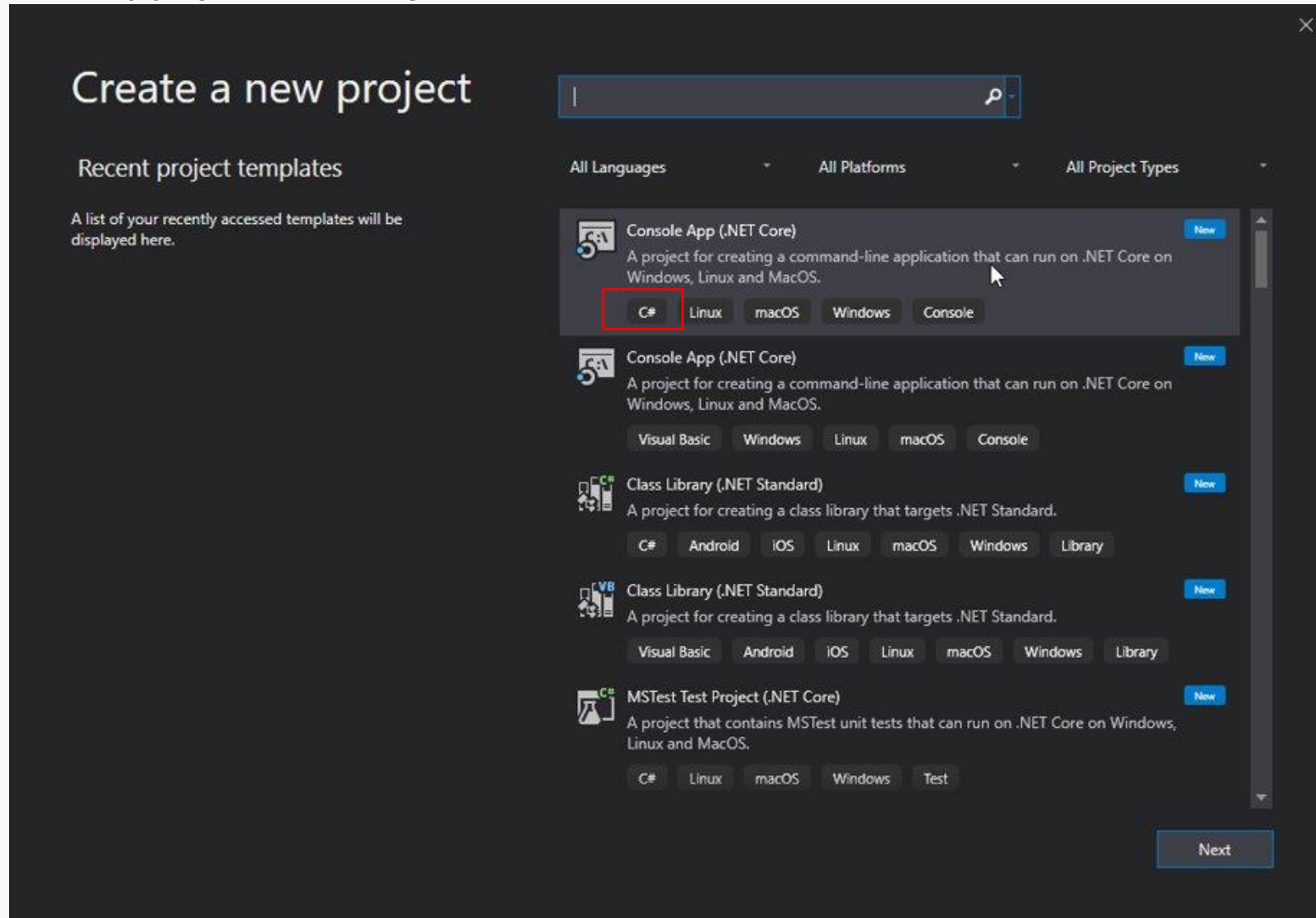
Configuration Visual Studio

click on the "Install more tools and features" button



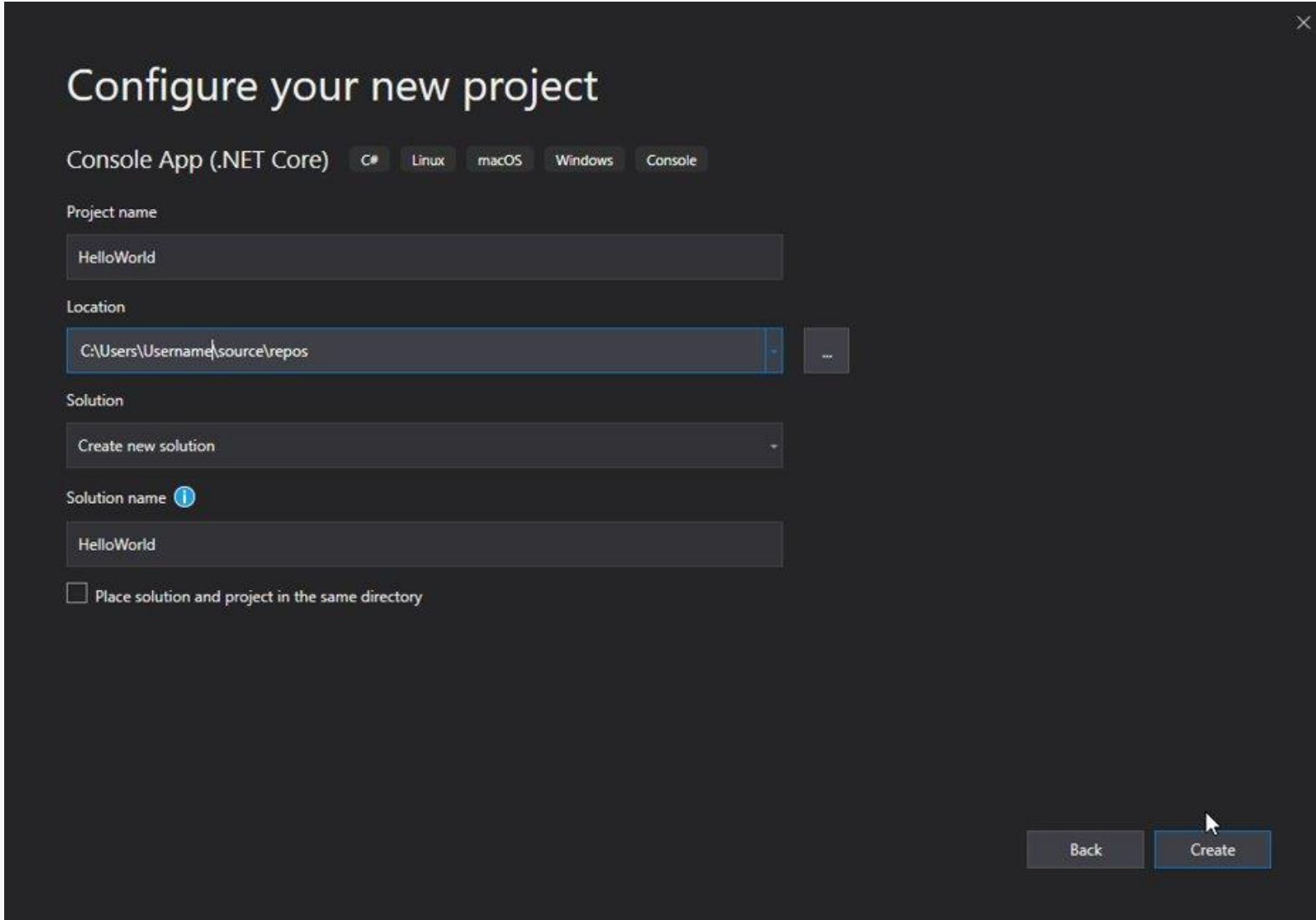
Configuration Visual Studio

Choose "Console App (.NET Core) with C#" from the list and click on the Next button



Configuration Visual Studio

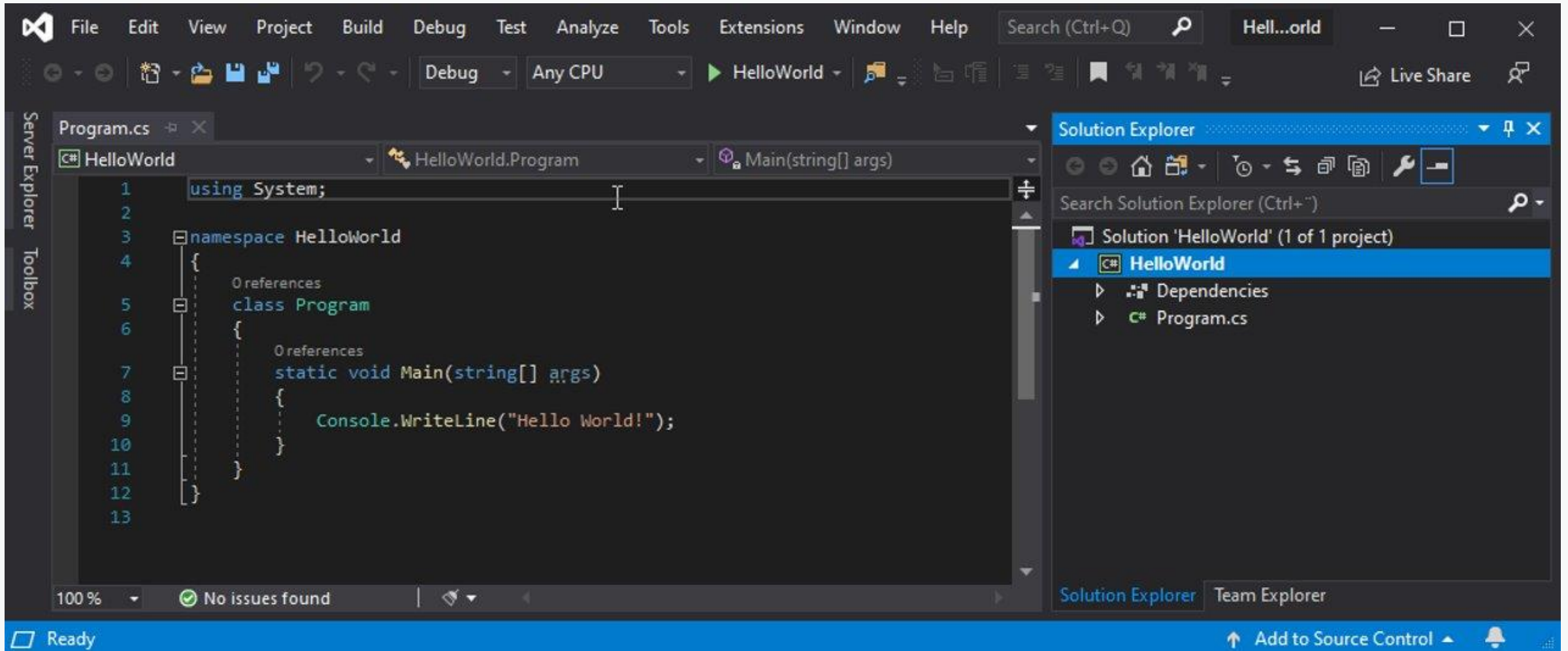
Enter a name for your project, and click on the Create button



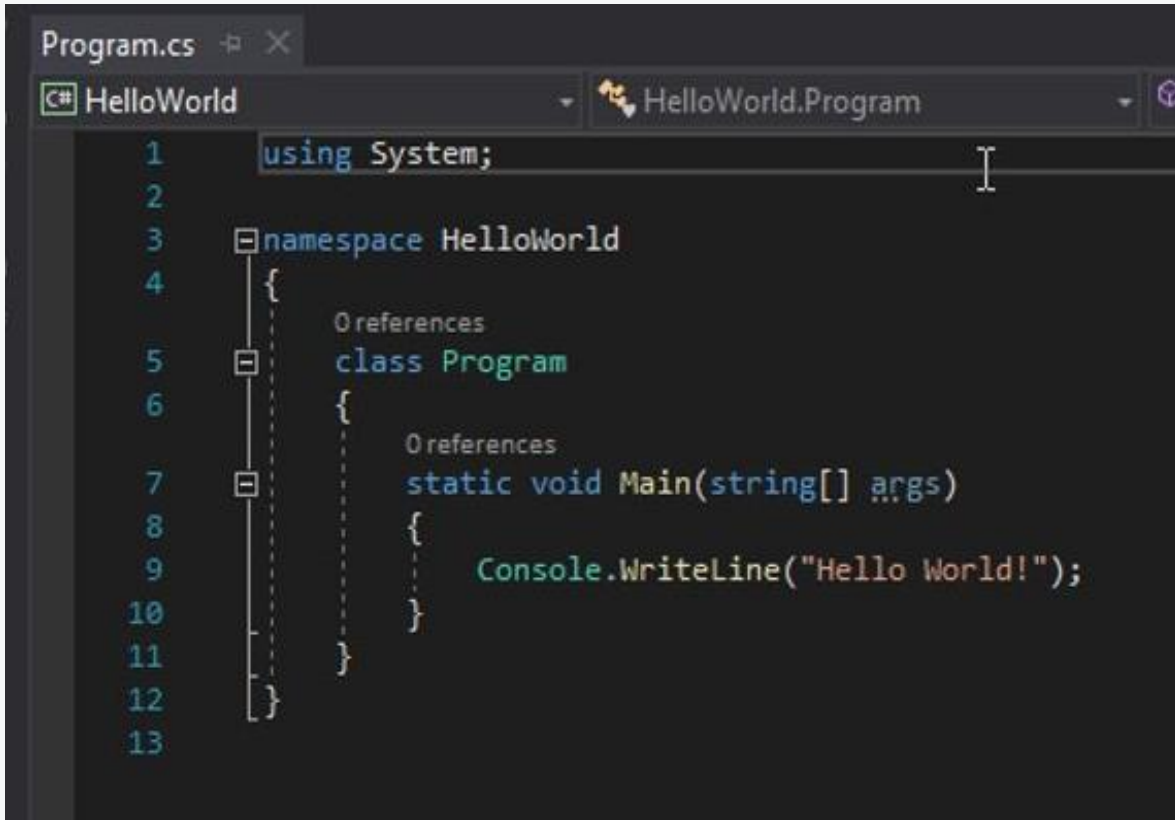
The screenshot shows the 'Configure your new project' dialog in Visual Studio. At the top, it says 'Configure your new project' with a close button (X) in the top right corner. Below this, there are tabs for 'Console App (.NET Core)', 'C#', 'Linux', 'macOS', 'Windows', and 'Console'. The 'C#' tab is selected. The 'Project name' field contains 'HelloWorld'. The 'Location' field contains 'C:\Users\Username\source\repos' with a browse button (three dots) to its right. The 'Solution' dropdown menu is set to 'Create new solution'. The 'Solution name' field, which has an information icon (i) to its left, also contains 'HelloWorld'. There is an unchecked checkbox labeled 'Place solution and project in the same directory'. At the bottom right, there are 'Back' and 'Create' buttons, with a mouse cursor hovering over the 'Create' button.

Configuration Visual Studio

Visual Studio will automatically generate a generic class for you



Configuration Visual Studio



```
1  using System;
2
3  namespace HelloWorld
4  {
5      class Program
6      {
7          static void Main(string[] args)
8          {
9              Console.WriteLine("Hello World!");
10         }
11     }
12 }
13
```

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

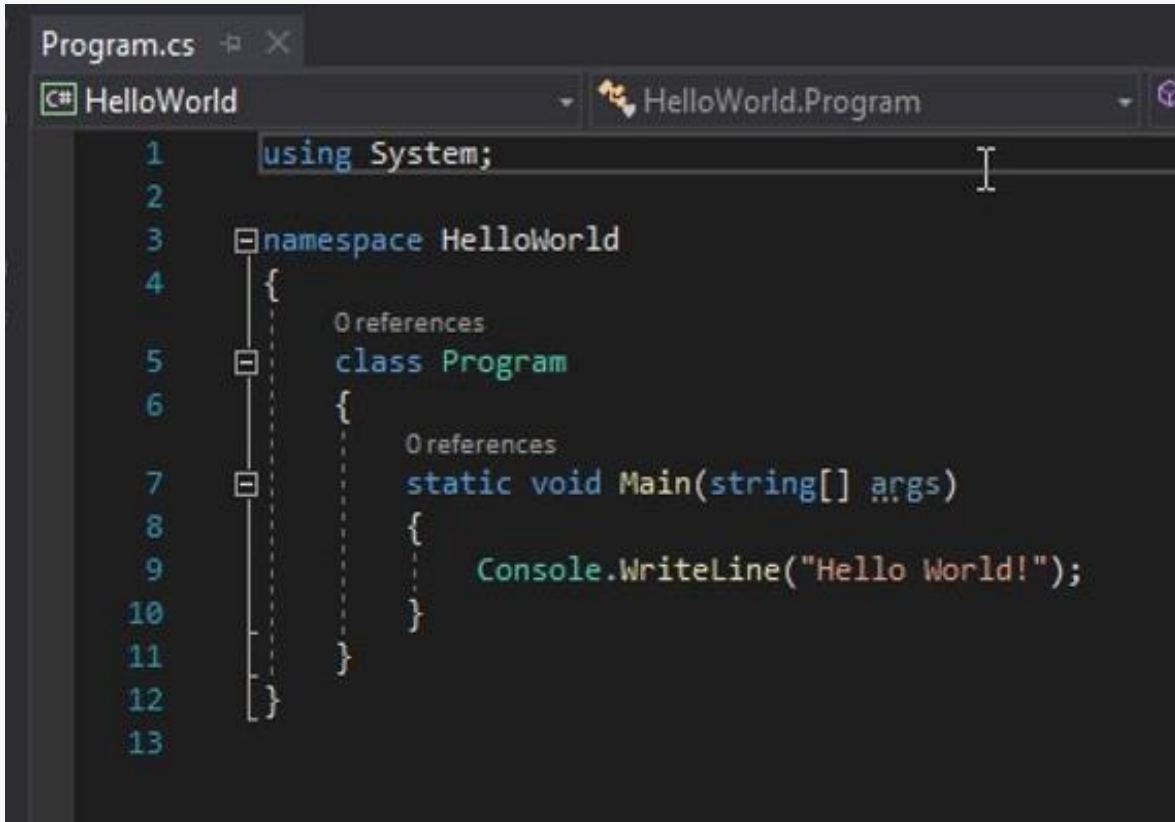
Line 2: A blank line. C# ignores white space. However, multiple lines makes the code more readable.

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. In our example, we named the class `Program`.

Configuration Visual Studio



```
1  using System;
2
3  namespace HelloWorld
4  {
5      class Program
6      {
7          static void Main(string[] args)
8          {
9              Console.WriteLine("Hello World!");
10         }
11     }
12 }
13
```

Line 7: Another thing that always appear in a C# program, is the **Main** method. Any code inside its curly brackets **{}** will be executed. You don't have to understand the keywords before and after Main. You will get to know them bit by bit throughout our lab session.

Line 9: **Console** is a class of the **System** namespace, which has a **WriteLine()** method that is used to output/print text. In our example it will output "Hello World!". If you leave out the **using System** line, you would have to write **System.Console.WriteLine()** to print/output text.

Note:

- Every C# statement ends with a semicolon **;**.
- C# is case-sensitive: "MyClass" and "myclass" has different meaning.

THANKS!