**Basics of C#.NET Programming**

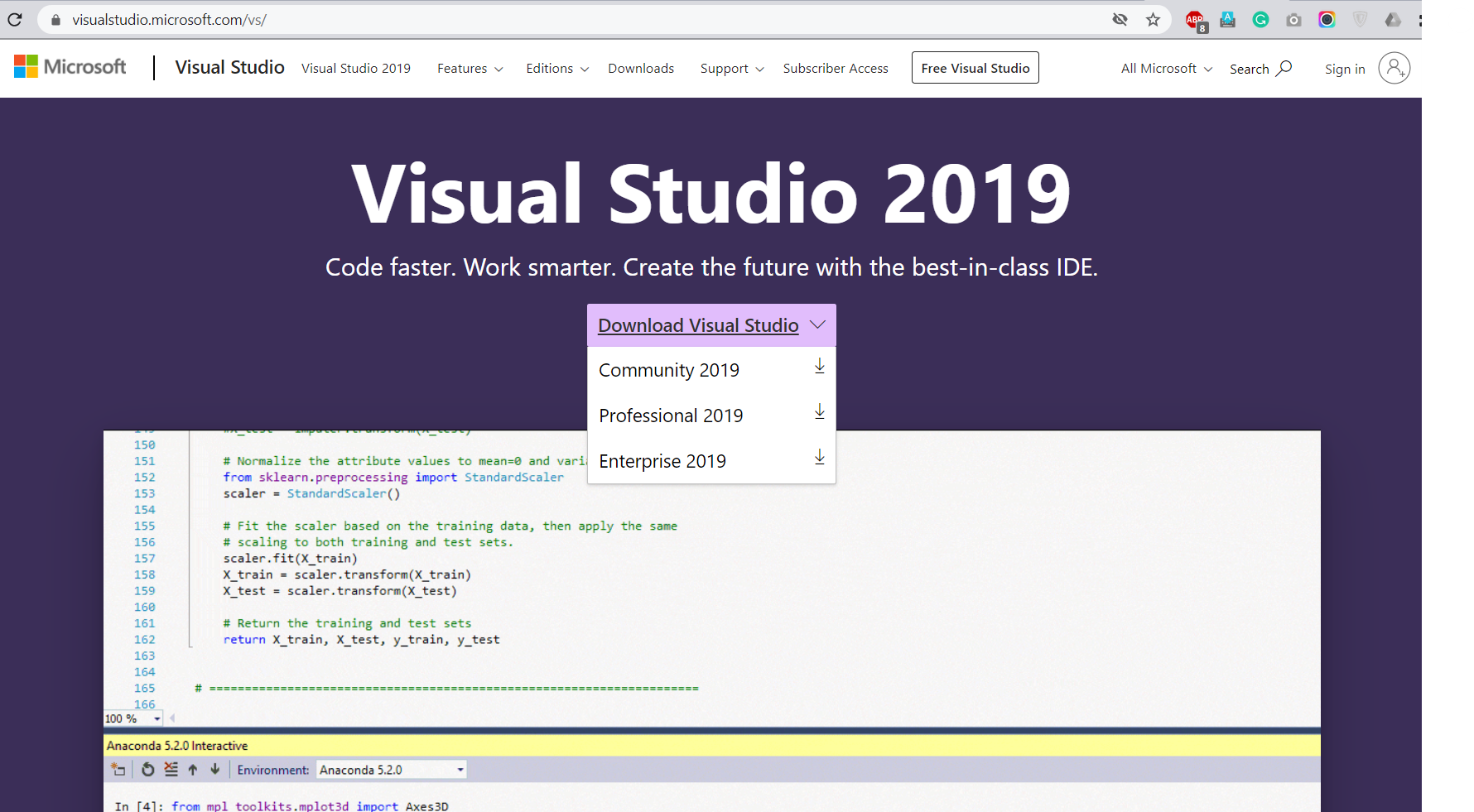
**(Part no. 2)**

# Installing Visual Studio:

Welcome to the Visual Studio IDE. As I already mentioned in the previous tutorial that we will be using Visual Studio as the IDE (Integrated Development Environment) for C#.NET as many of the .NET libraries and .NET itself works at its best in this environment. Visual Studio was also developed by Microsoft. It supports about 36 programming languages whereas on the other hand its compilers, code editor, and debugger support nearly all kinds of programming languages. There are three versions of Visual Studio actively used every day around the globe and these are;

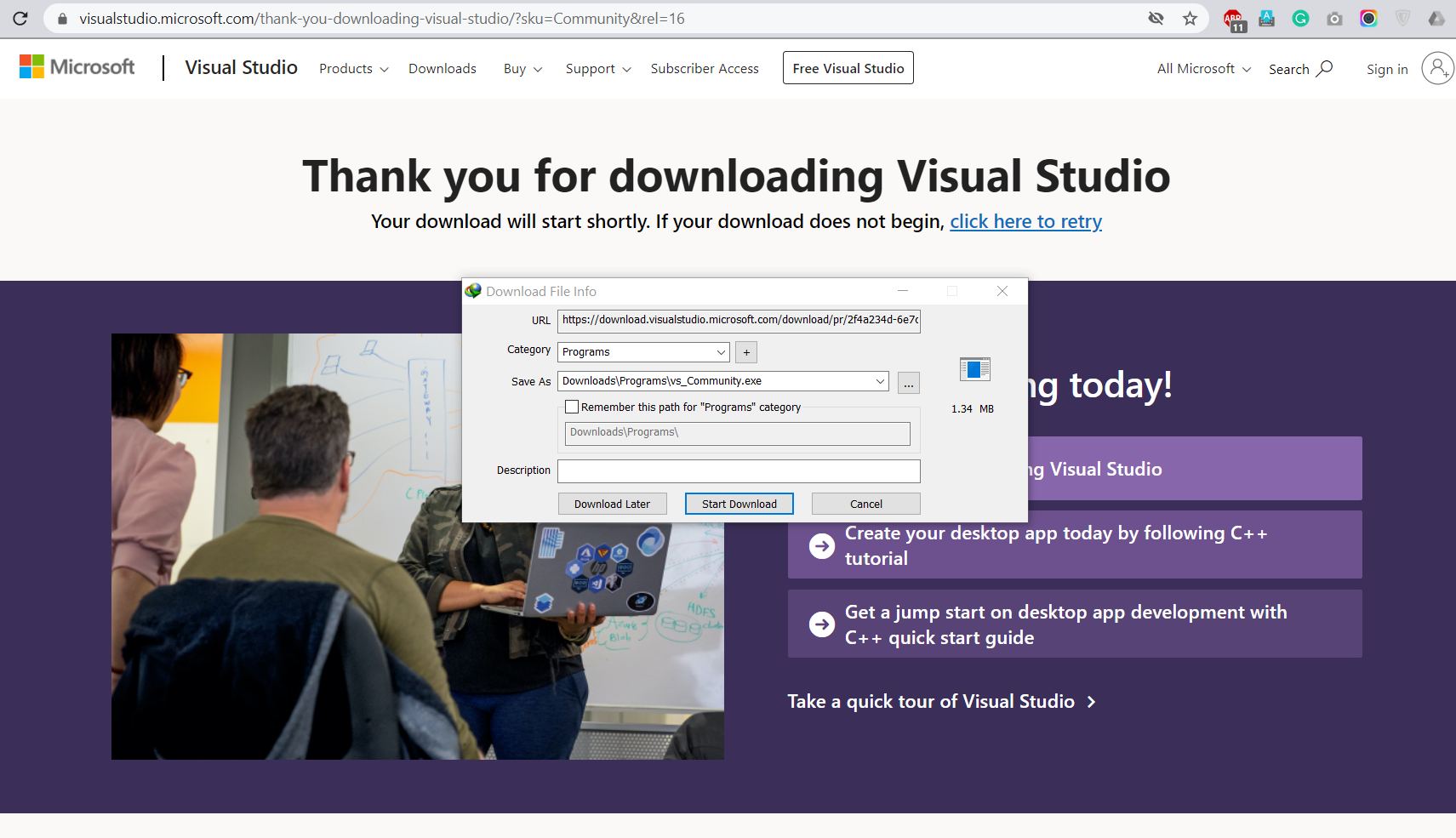
* Visual Studio Community, which is free of any cost and mostly used by the students, open-source programmers, developers, and freelancers.
* Visual Studio Professional, which comes with some price and is mostly used by the professional programmers and coders. It is enriched with premium APIs and lots of other configuration settings that are not available in the Community version.
* Visual Studio Enterprise, which comes at a high price mostly used in the enterprise corporate sector, reputable companies, and a small professional team of developers. It has agile support integration through which different teams can develop and deliver the user end products with a minimal amount of time and effort.

That’s it for that much explanation about the visual studio and let's jump right into its installation. For installing the Visual Studio Community version on your system you can click [here](https://visualstudio.microsoft.com/vs/). That will lead you directly to the main website of Visual Studio IDE where you will download the Community version. As in the figure below

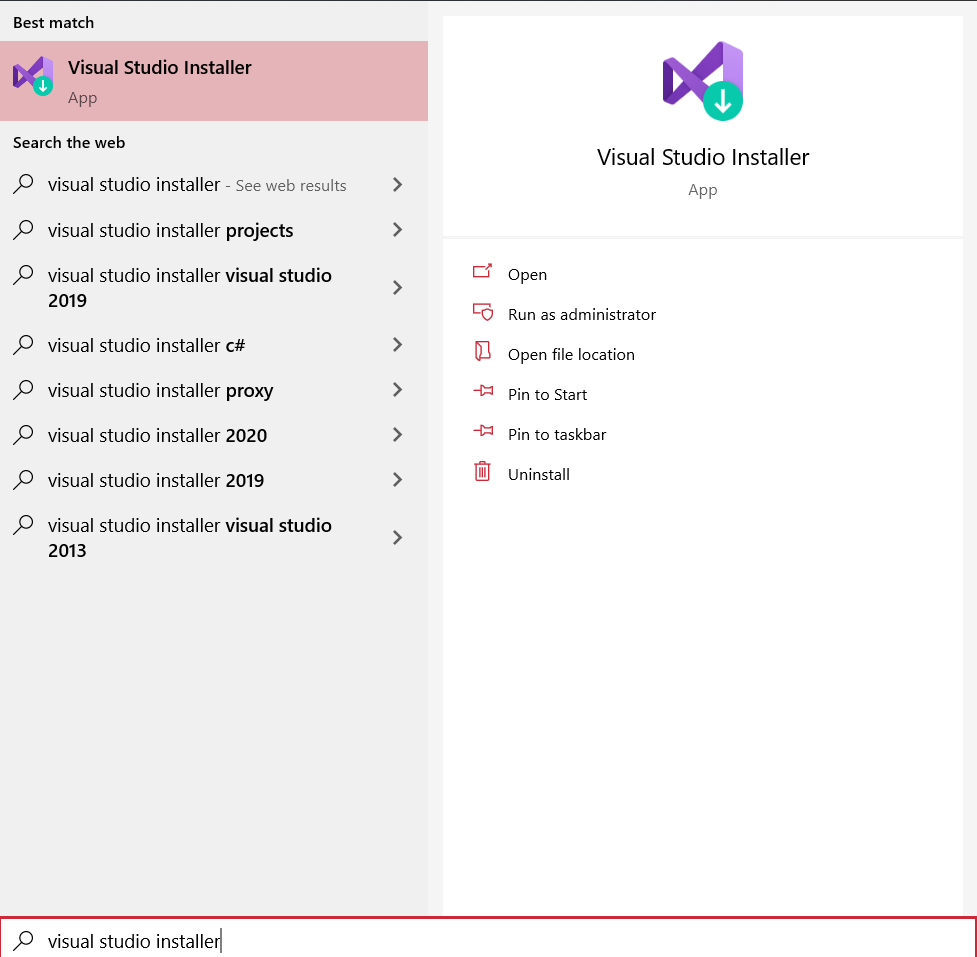


But if you don’t find it at ease then you can type ‘visual studio IDE’ in the search bar of your browser and hit enter, the very first two search results that Google will show you just click on them and they will lead you right on the above figure page.

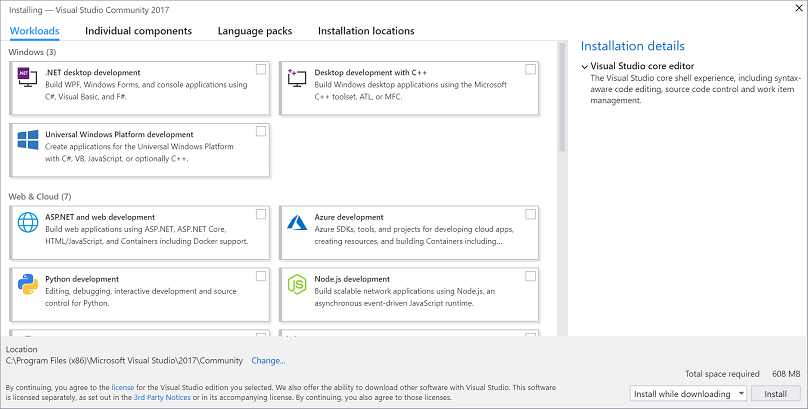
After that, you have to click on the Visual Studio Community Version, which will lead you to the next site greeting ‘thankyou for downloading Visual Studio’ after which you will a see prompt of your third party downloader like IDM or if not installed then it will download in the background of the Chrome browser. Like the following



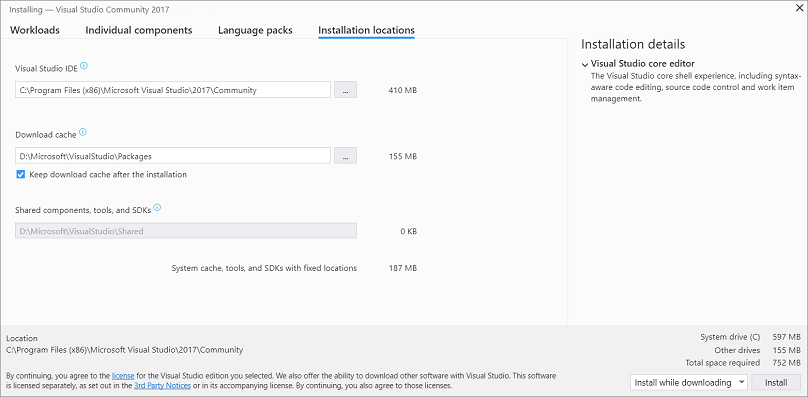
After your download completes open that application and installs it. When the installation is completed go on the search bar of your computer and search for a visual studio code installer. Like



And open it up and you will see this interface



Here you just have to check the option of .NET desktop Development and then proceed with installing it will take about 4 to 8 hours to install on your system, depending upon the bandwidth of your internet connection. If it's greater then it will do within 2 hours max. Just take care of one thing to install in the main directory drive where your operating system resides like if it's in SSD drive then make sure to install it there. Because it will help in the loading and speed of the project.



The file size may range from 6 GB to 10 GB don’t let the above figure stop you from following my instructions cause it's just an illustration of where it should be installed. I’m sure by now it will be very easy for you guys but let's jump right into opening the Visual Studio and making our very first console application.

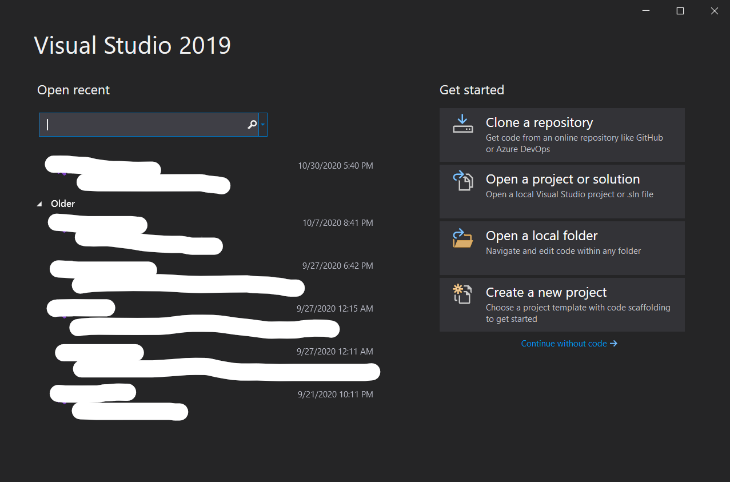
# Frist Console Application:

You guys might be wondering that why did I start with a Console Application? Because in the beginning or for the beginners we try to practice and build our foundations or basics of programming on a Console application. For the next 10-20 lectures, you guys and I will be seeing our result only on a black screen. But trust me when it's complete, we will move on towards the desktop application development by using WPF, MVC, and other .NET modules. Let's open up our Visual Studio Community 2019 version and start building our very first project.

Go in the search bar and type ‘Visual Studio’ and the search will appear, click on it and you will see

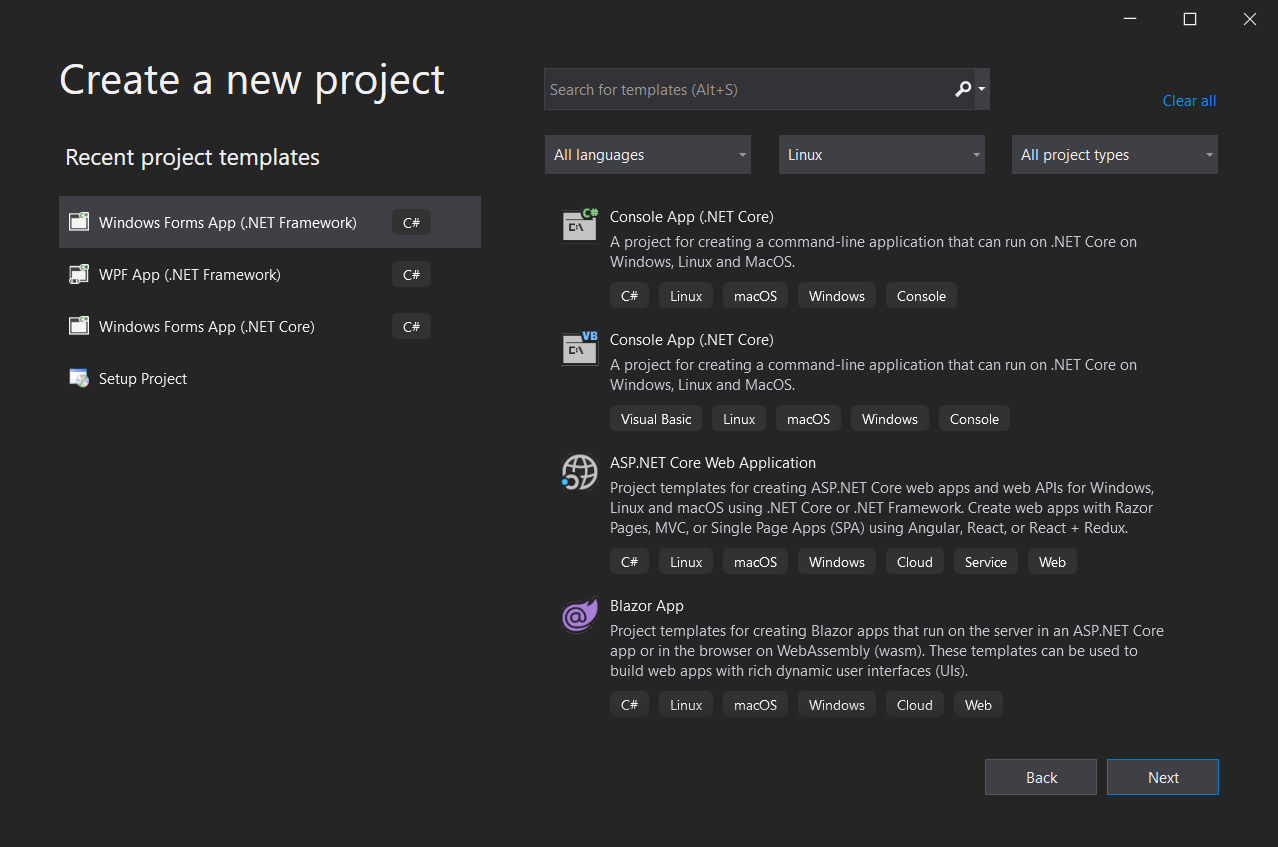


It will take some loading time in the very beginning and you should give it some time to load components and packages etc. after that you will this main window

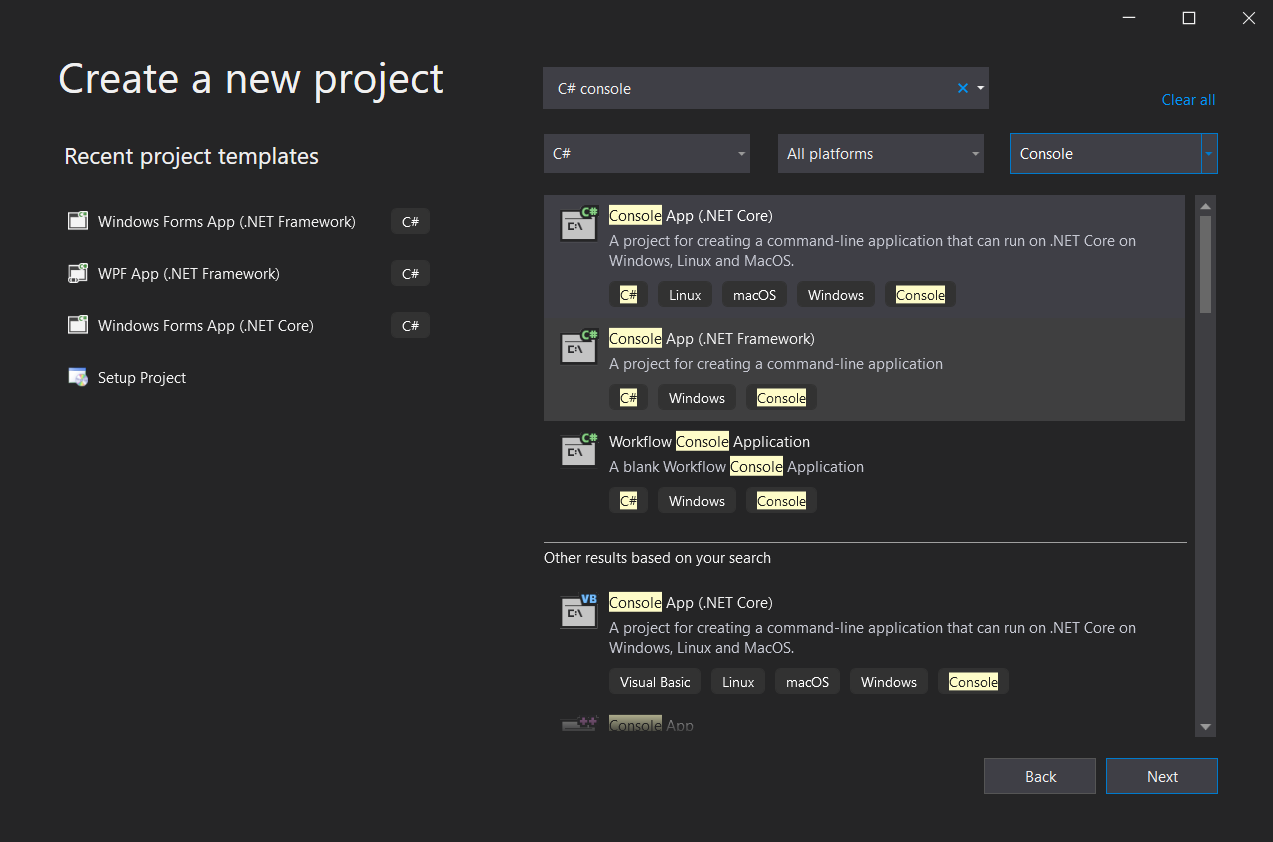


Sorry for hiding some classified information😜 if you know what I mean😉

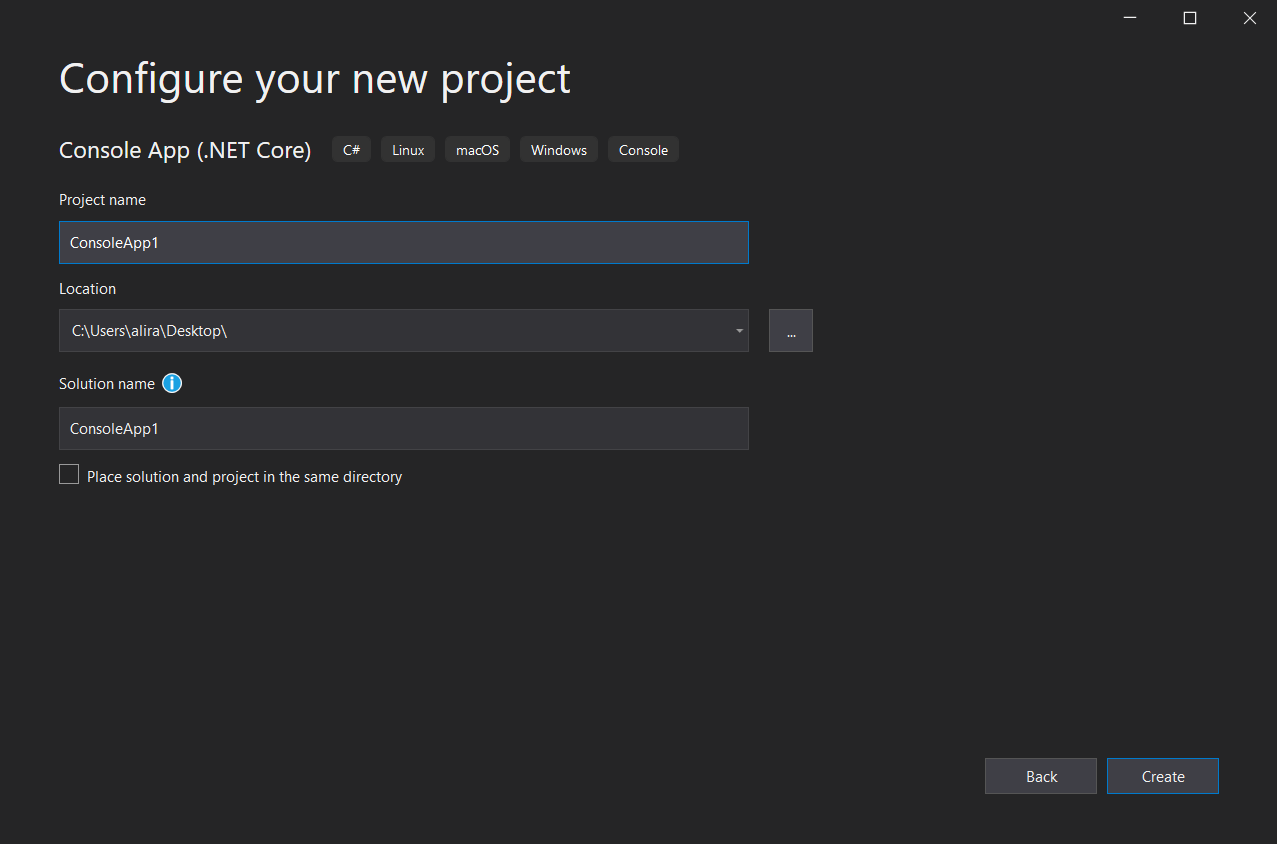
Now let's move on to creating our very first project on C# as a console application project by clicking on the create a new project button in the above figure which will take us to the next window as



Type in the search bar ‘C# console’ and you will see the following results



In the above figure, you will two option one is (.NET Core) and (.NET Framework) you will choose the (.NET Framework) option and click on the next



Here you can name your project in your way or just follow the instructions like typing ‘ConsoleApp1’. Here’s an important thing to keep in mind about naming conventions before naming the project. You can only use the “Camal Case” or the “Pascal Case”. You will be surprised and might be in great shock to hear these two terminologies out of nowhere. So let's discuss them before proceeding further.

## Camel Case

C# is also a very case sensitive language too. So, you have to keep in mind some rules before naming the project. Let's discuss this case. In Camel Case, we name the project just like the Camal’s hump, like starting with small alphabets and then capital alphabets in between. Following are some useful examples.

* age
* classRoom
* area350

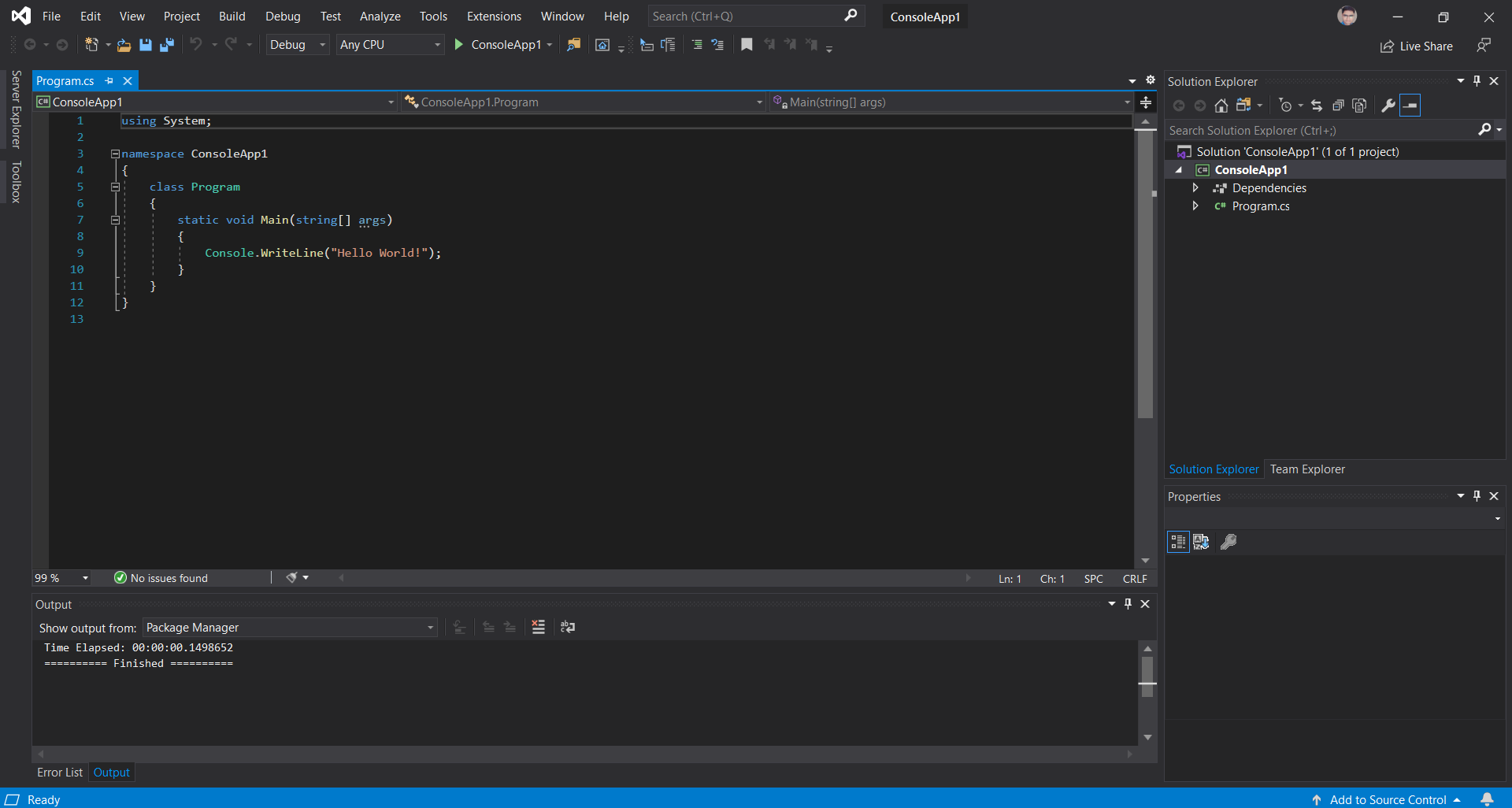
Yes, you can also use numbers in the naming of the projects. But keep in mind that you can’t start with the numbers before the alphabets. Like “2app” etc.

## Pascal Case

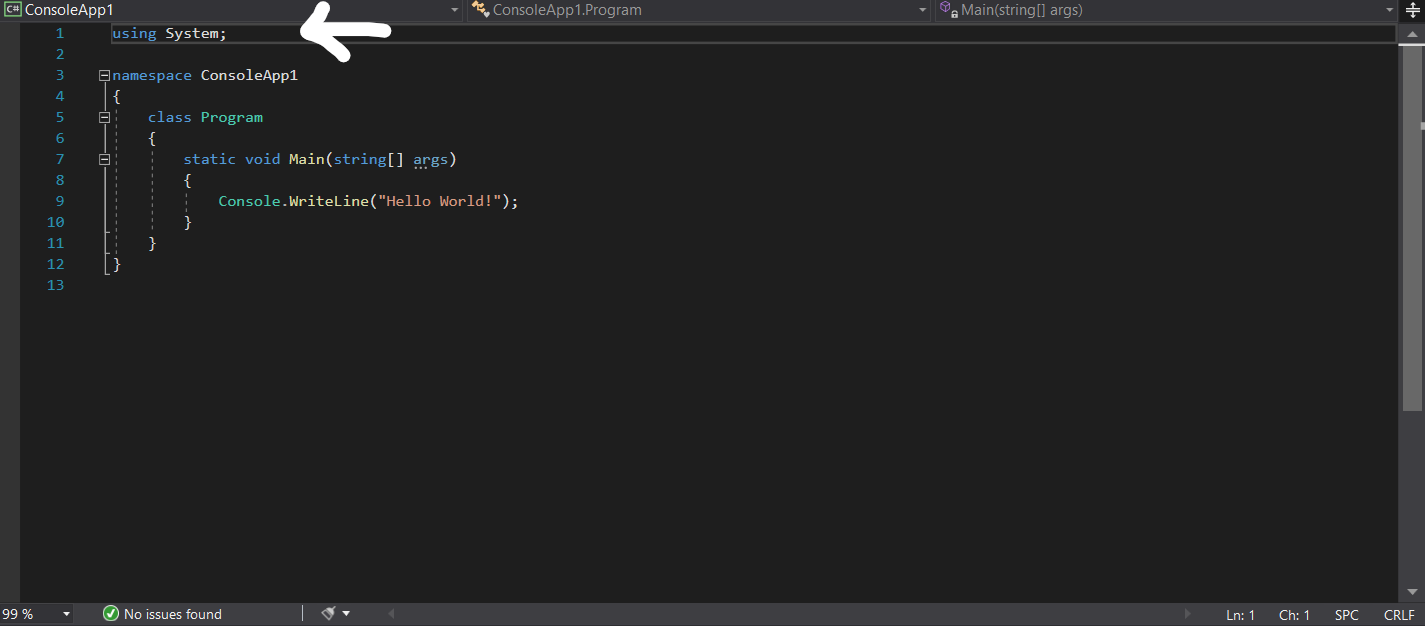
Other than that of Camal Case, Pascal case is more used and we will also be using this naming convention in the rest of our tutorials. It starts from Capital Letters and like Camal Case you can also add numbers after the alphabets. Like

* Standard form
* ClassRoom
* Discount50 etc

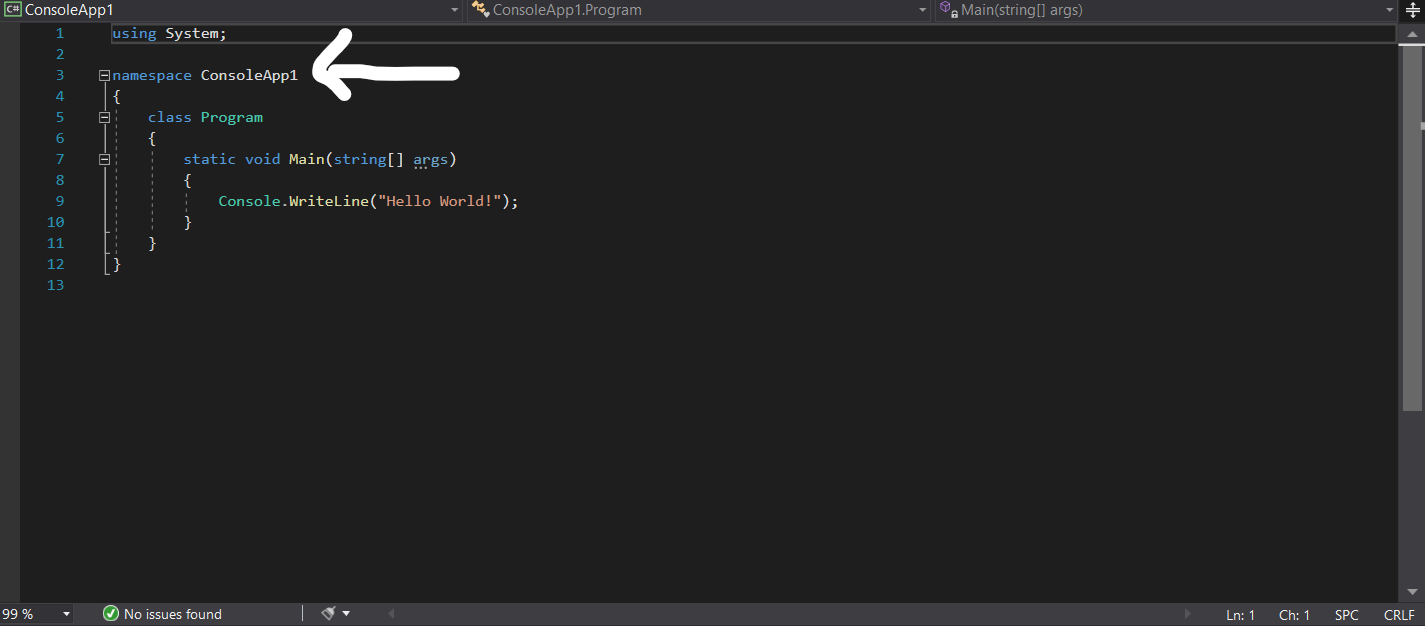
Let's move with our project and see what happens after we click the create button



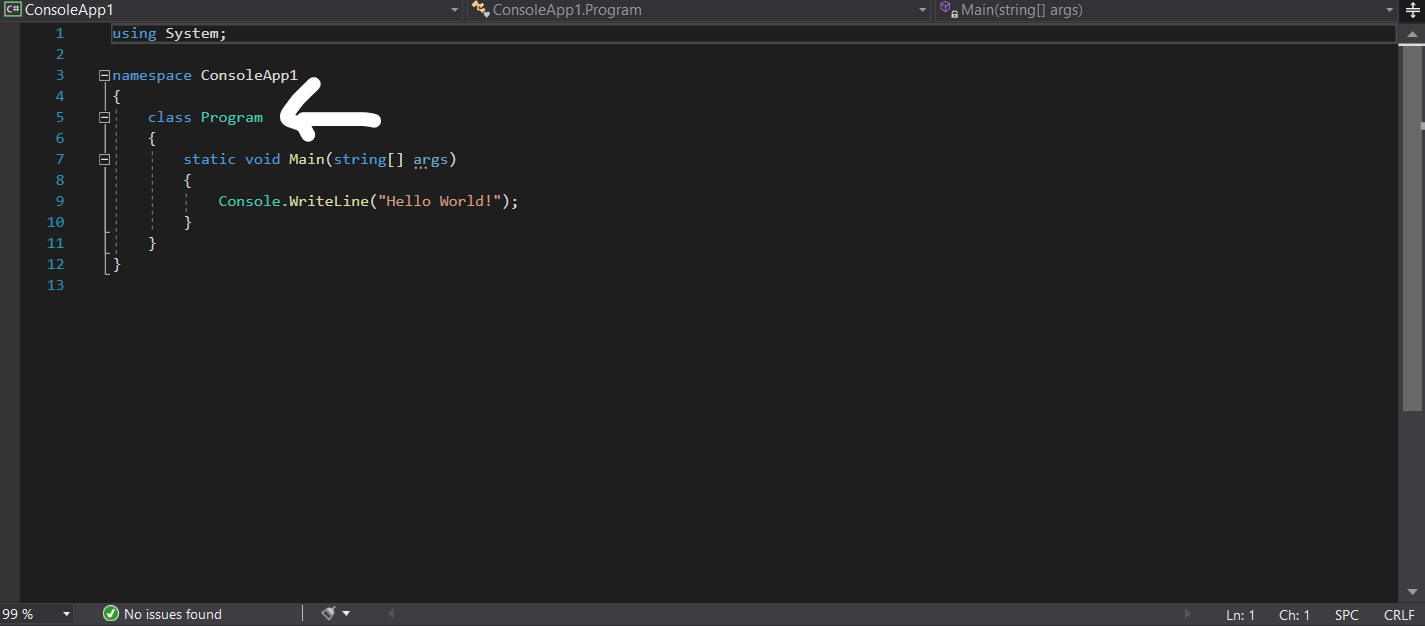
As you can see that a code is already present there. This is one of the features Visual Studio discussed in the previous tutorial and it's called the “boilerplate” code. Let's start breaking into this code



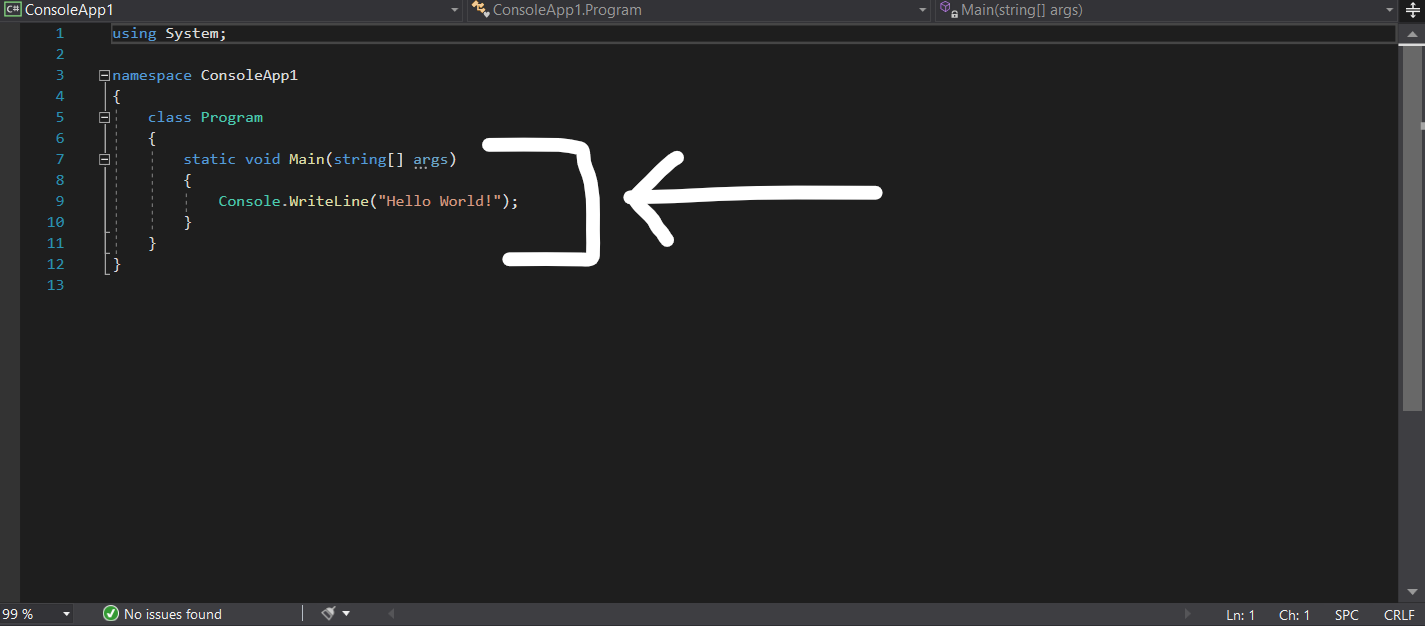
In the above figure as you can see the ‘using System;’ These are the collections and libraries or the namespaces that will help the program to run. Like if we build a grade book or a calculator in our later tutorials then we will go for the different other namespaces like system.collection, system.generic, etc. Every namespace includes its functions and classes like mathematical operations, statistics, and others. Let's move to the next line.



In the above figure, you can see that our project name is also stated before the namespace and you will wonder that Ali Raza what is this happening? Then let me clear you that this is our namespace called the ConsoleApp1, which I stated before that the namespaces are also the classes and libraries. Or you can also call it the meta namespace (namespace of the namespace). Let's see what another line does



In the above one, you will be seeing ‘class program’ which refers to the program that is the program belonging to a class. Furthermore, this one will be discussed in our later tutorials in the future. Let's discuss the main part of this project now.

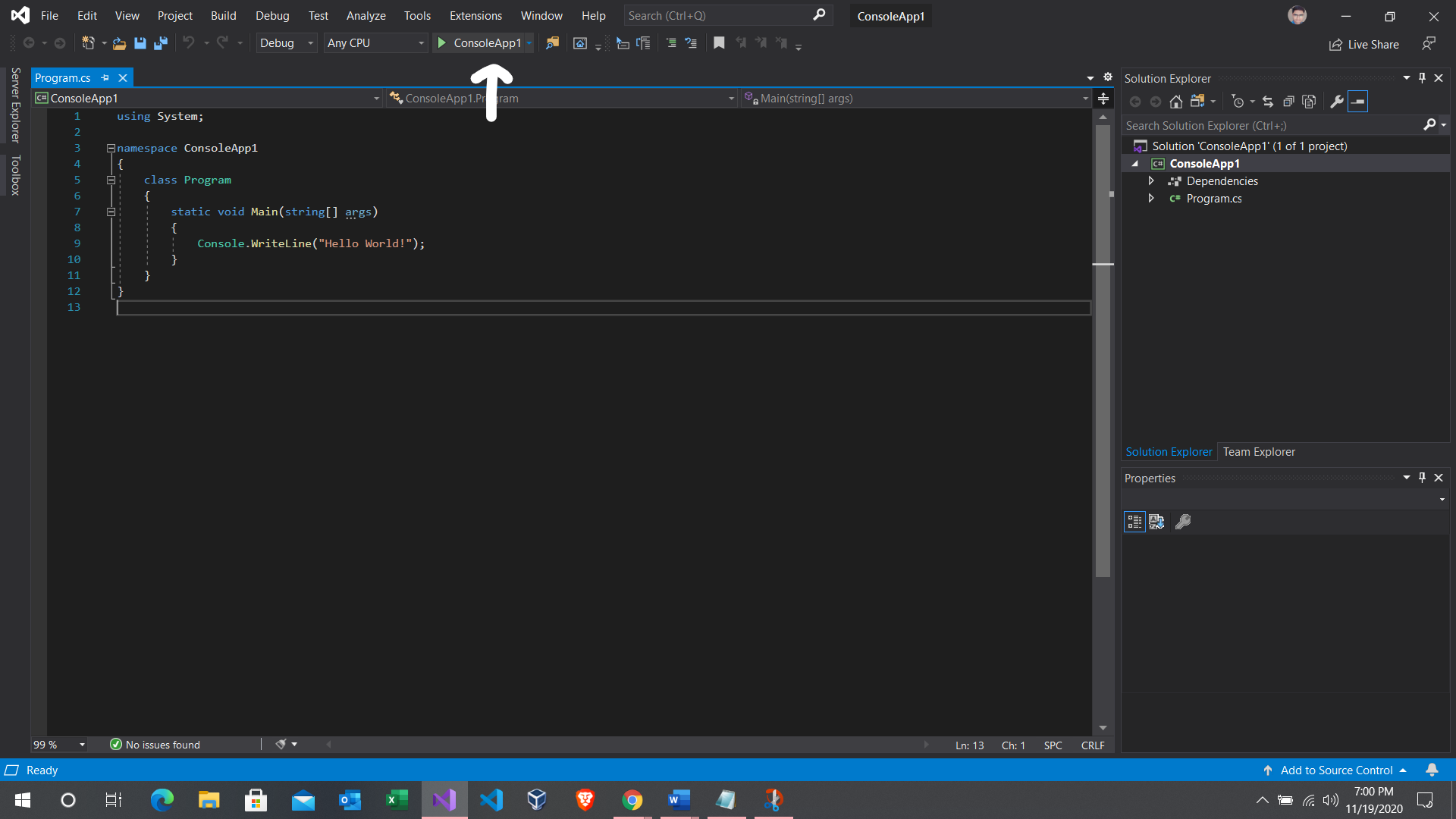


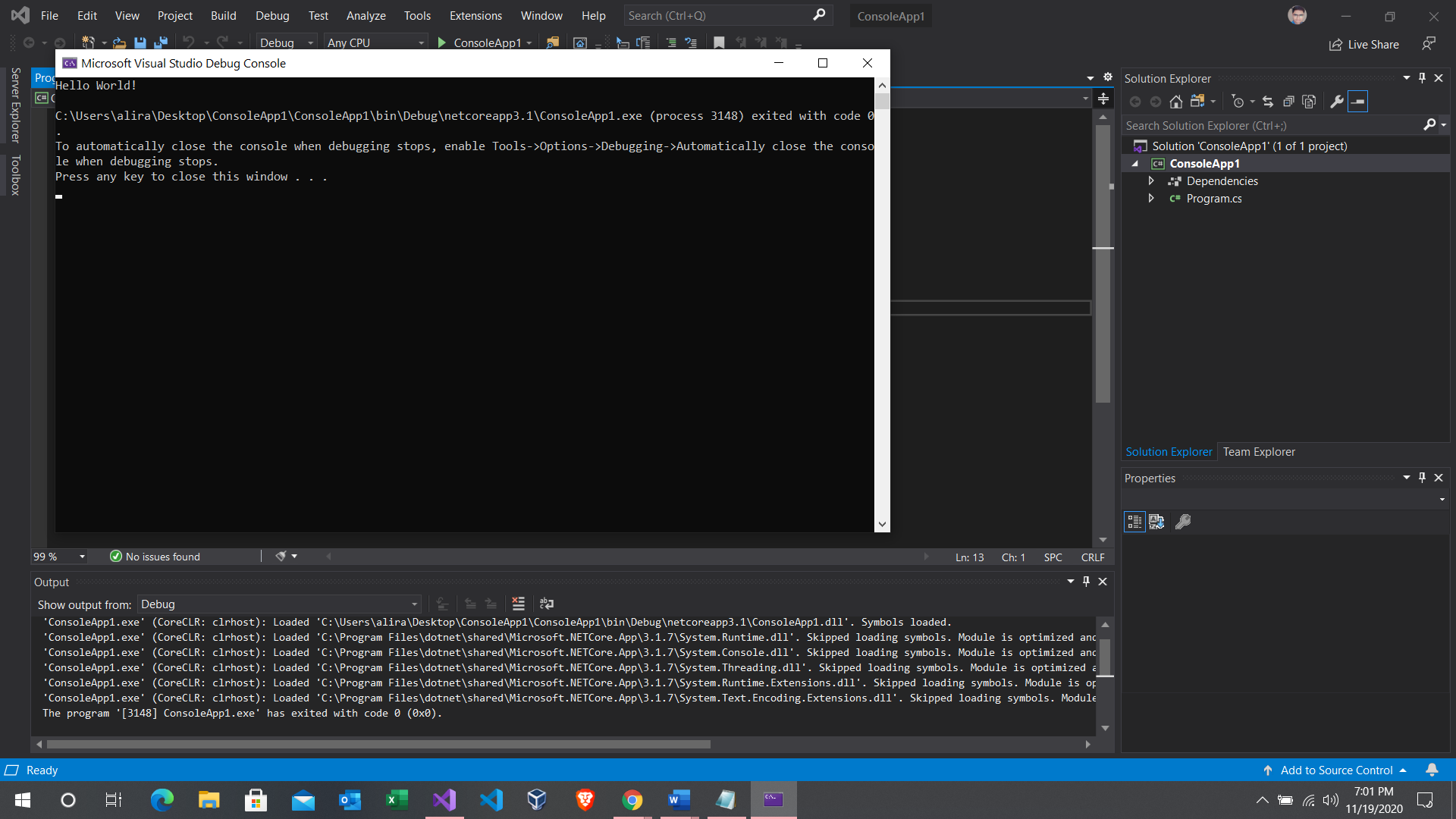
As you can see that I have highlighted the whole region in this one and that is because these all things are related and connected, even every element if the project does.

In the first line, the ‘static’ word shows that C# is a static language. Whereas the ‘Main’ is the function and it refers to the main entrance into the program where the major part of the project is coded. It helps the compiler to distinguish between the different blocks of codes. Whereas the string[] and ages are the parameters that will be discussed in later tutorials.

Let's talk about the inside code, ‘Console’ tells the compiler using the namespace ‘System’ that the function that is about to take place is related to the console and the result should be displayed on the console screen. In this case, it’s the “Hello World!”.

Let's run this program and see what happens by clicking on the indicated button.





As you can see that the code is compiled successfully. And the “Hello World!” is displayed on the top of the console screen. Ignore the rest because it's only the path where the project is present and some instructions which only comes in the very first project compilation.

That’s it for today's lecture I hope you have learned a lot. Stay tuned for further lectures in the future. 😇