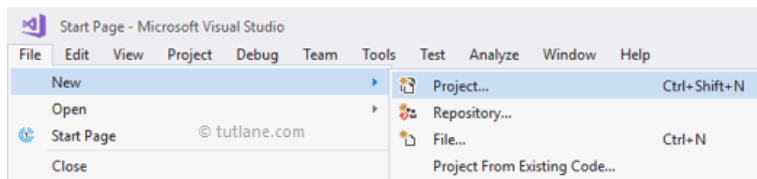# C# Hello World Program Example

Using Visual Studio, you can easily create a Hello World Program or Console Application in C# programming language based on our requirements.
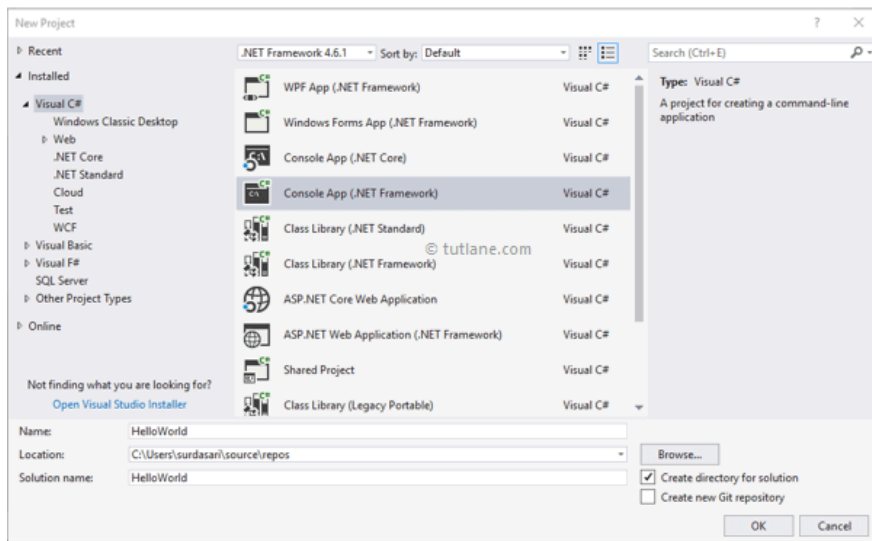
In the previous chapter, we explained how to Download and Install Visual Studio on Windows Machine (/tutorial/csharp/csharp-setup-development-environment). If you are not installed a visual studio, follow the instructions to install visual studio (/tutorial/csharp/csharp-setup-development-environment); otherwise, open your visual studio.
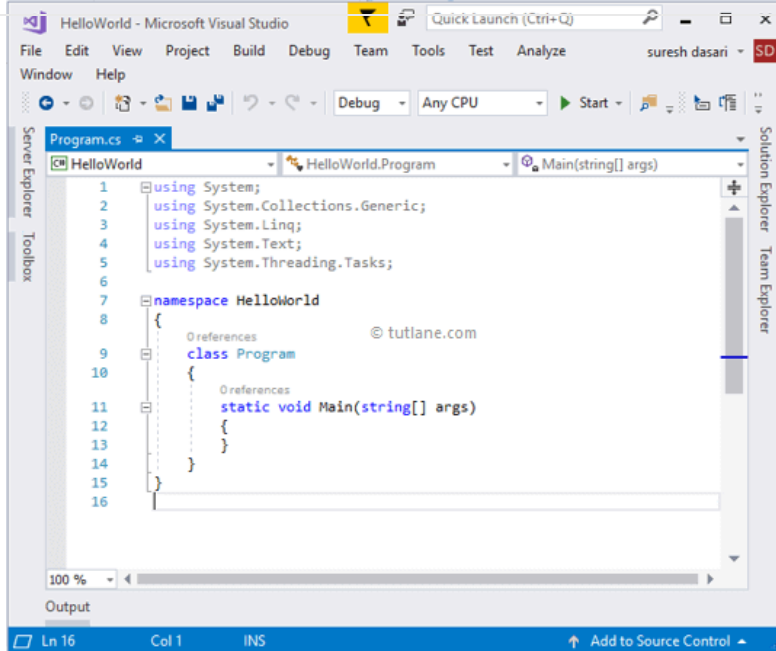
Create a C# Console Application

To create a new application in visual studio, go to the Menu bar, select **File → New →** select a **Project** like as shown below.

Once you click on **Project**, a new popup will open in that select **Visual C#** from the left pane and choose **Console App**. In the **Name** section, give any name for your project and select an appropriate **Location** path to save your project files, and click **OK** like as shown below.

Once you click on the **OK** button, a new console application will be created like as shown below. In case the **Program.cs** file not opened in your code editor, open **Solution Explorer** menu on the right side and open your **Program.cs** file code by double-clicking it.

If you observe the above image, by default the application contains a **Main()** method because the console applications in the c# programming language will always start from the **Main()** method of the Program class.

C# Hello World Program Example

Now replace your **Program.cs** file code as shown below to display the "**Hello World**" message.
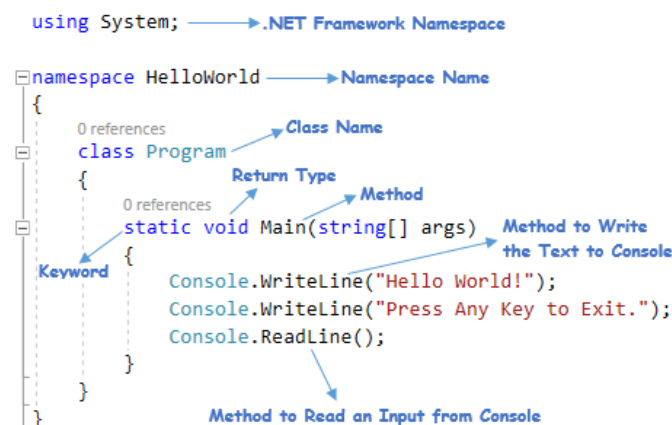
```csharp
using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            Console.WriteLine("Press Enter Key to Exit.");
            Console.ReadLine();
        }
    }
}
```

If you observe the above code, we used many parameters to implement the "**Hello World**" program in c#. In the next section, we will learn all the parameters in a detailed manner.

Explanation of C# Hello World Program

The following diagram will illustrate the parameters that we used in our c# program in a detailed manner.

We will go through each step of our c# program and learn each parameter in a detailed manner.

using System;

Here, `using System` is the .NET Framework library namespaces, and we used `using` keyword to import `system` namespace to use existing class methods such as **WriteLine()**, **ReadLine()**, etc. By default, the **.NET Framework** provides a lot of namespaces to make the application implementation easy.

namespace HelloWorld

Here, `namespace HelloWorld` is the main namespace (/tutorial/csharp/csharp-namespaces-with-examples) of our application, and by default, our application classes will be a part of it.

class Program

Here, `class Program` is used to define a class (**Program**) in the namespace (**Helloworld**). The class (**Program**) will contain all the variables, methods, etc., and we can define more than one class in same namespace based on our requirements.

static void Main(string[] args)

Here, `static void Main(string[] args)` is used to define a method in our class (**Program**).

- The keyword `static` tells us that the **main** method can be accessible without instantiating the class (**Program**).
- Another keyword `void` tells us that what this method should return.
- The name **Main** will refer to the name of our class method (**Program**). The **Main()** method is the entry point of our console application.
- After the name (**Main**) of a method, we defined a set of parameters within parentheses. Here our method takes only one parameter, called `args` and it is useful to send command-line arguments as text strings for our main method.

Console.WriteLine() / ReadLine()

Here, `Console.WriteLine()` and `Console.ReadLine()` methods are used to write a text to the console and read the input from the console.

The `Console` is a class of .NET Framework namespace `System` and WriteLine() and ReadLine() are the methods of `Console` class.
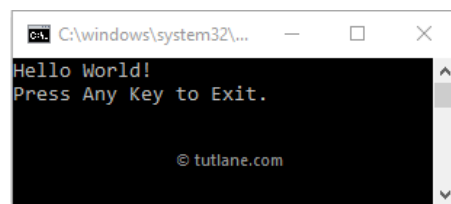
> ❶ In **c#** programming language, every statement or line must end with a **semicolon (;)**.

Compile and Run C# Hello World Program

To see our Hello World C# Program's output, you need to compile and run the application by pressing either `Ctrl + F5` or click on the **Start** option in the menu bar like as shown below.

Once you click on the **Start** option or `Ctrl + F5`, our program will get compiled and show the result as shown below.

This is how you can create and execute the c# programming language applications using visual studio based on our requirements.