About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: csharp-language

It is an unofficial and free C# Language ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official C# Language.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with C# Language

Remarks

C# is a multi-paradigm, C-descendant programming language from Microsoft. C# is a managed language that compiles to CIL, intermediate bytecode which can be executed on Windows, Mac OS X and Linux.

Versions 1.0, 2.0 and 5.0 were standardized by ECMA (as ECMA-334), and standardization efforts for modern C# are underway.

Versions

Version	Release Date
1.0	2002-01-01
1.2	2003-04-01
2.0	2005-09-01
3.0	2007-08-01
4.0	2010-04-01
5.0	2013-06-01
6.0	2015-07-01
7.0	2017-03-07

Examples

Creating a new console application (Visual Studio)

1. Open Visual Studio
2. In the toolbar, go to **File** → **New Project**
3. Select the **Console Application** project type
4. Open the file `Program.cs` in the Solution Explorer
5. Add the following code to `Main()`:

```
public class Program
{
    public static void Main()
    {
        // Prints a message to the console.
```

```
        System.Console.WriteLine("Hello, World!");

        /* Wait for the user to press a key. This is a common
           way to prevent the console window from terminating
           and disappearing before the programmer can see the contents
           of the window, when the application is run via Start from within VS. */
        System.Console.ReadKey();
    }
}
```

6. In the toolbar, click **Debug** -> **Start Debugging** or hit **F5** or **ctrl + F5** (running without debugger) to run the program.

Live Demo on ideone

Explanation

- `class Program` is a class declaration. The class `Program` contains the data and method definitions that your program uses. Classes generally contain multiple methods. Methods define the behavior of the class. However, the `Program` class has only one method: `Main`.
- `static void Main()` defines the `Main` method, which is the entry point for all C# programs. The `Main` method states what the class does when executed. Only one `Main` method is allowed per class.
- `System.Console.WriteLine("Hello, world!");` method prints a given data (in this example, `Hello, world!`) as an output in the console window.
- `System.Console.ReadKey()`, ensures that the program won't close immediately after displaying the message. It does this by waiting for the user to press a key on the keyboard. Any key press from the user will terminate the program. The program terminates when it has finished the last line of code in the `main()` method.

Using the command line

To compile via command line use either `MSBuild` or `csc.exe` *(the C# compiler)*, both part of the Microsoft Build Tools package.

To compile this example, run the following command in the same directory where `HelloWorld.cs` is located:

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\csc.exe HelloWorld.cs
```

It can also be possible that you have two main methods inside one application. In this case, you have to tell the compiler which main method to execute by typing the following command in the **console**.(suppose Class `ClassA` also has a main method in the same `HelloWorld.cs` file in

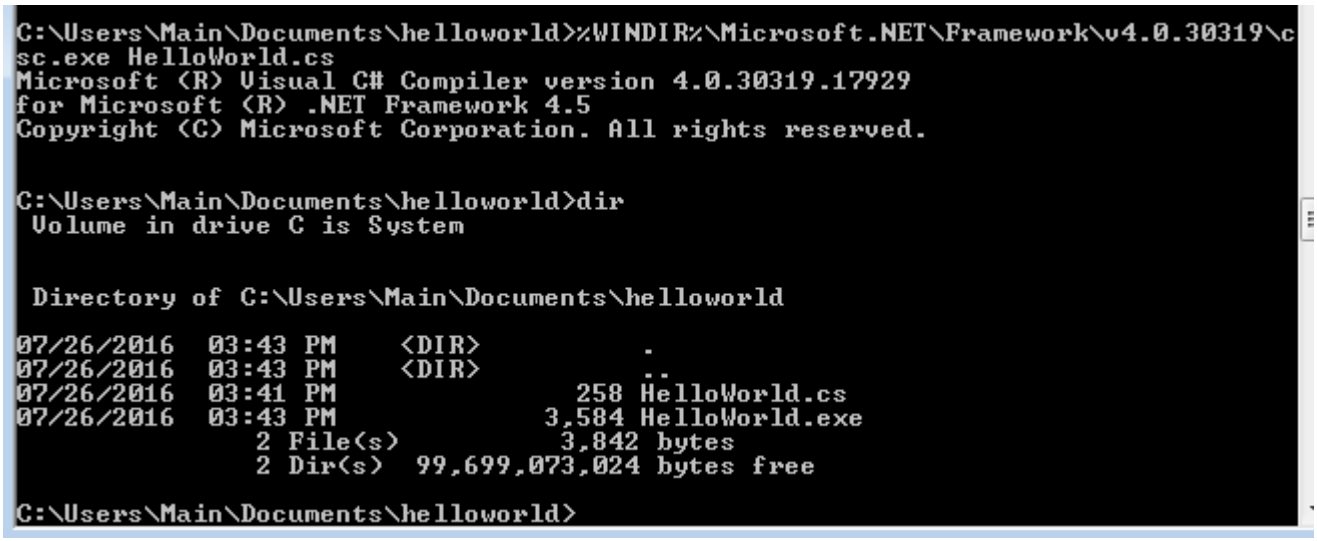
HelloWorld namespace)

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\csc.exe HelloWorld.cs /main:HelloWorld.ClassA
```

where HelloWorld is namespace

***Note**: This is the path where **.NET framework v4.0** is located in general. Change the path according to your .NET version. In addition, the directory might be **framework** instead of **framework64** if you're using the 32-bit .NET Framework. From the Windows Command Prompt, you can list all the csc.exe Framework paths by running the following commands (the first for 32-bit Frameworks):*

```
dir %WINDIR%\Microsoft.NET\Framework\csc.exe /s/b
dir %WINDIR%\Microsoft.NET\Framework64\csc.exe /s/b
```



There should now be an executable file named `HelloWorld.exe` in the same directory. To execute the program from the command prompt, simply type the executable's name and hit `Enter` as follows:

```
HelloWorld.exe
```

This will produce:

> Hello, world!

You may also double click the executable and launch a new console window with the message "**Hello, world!**"