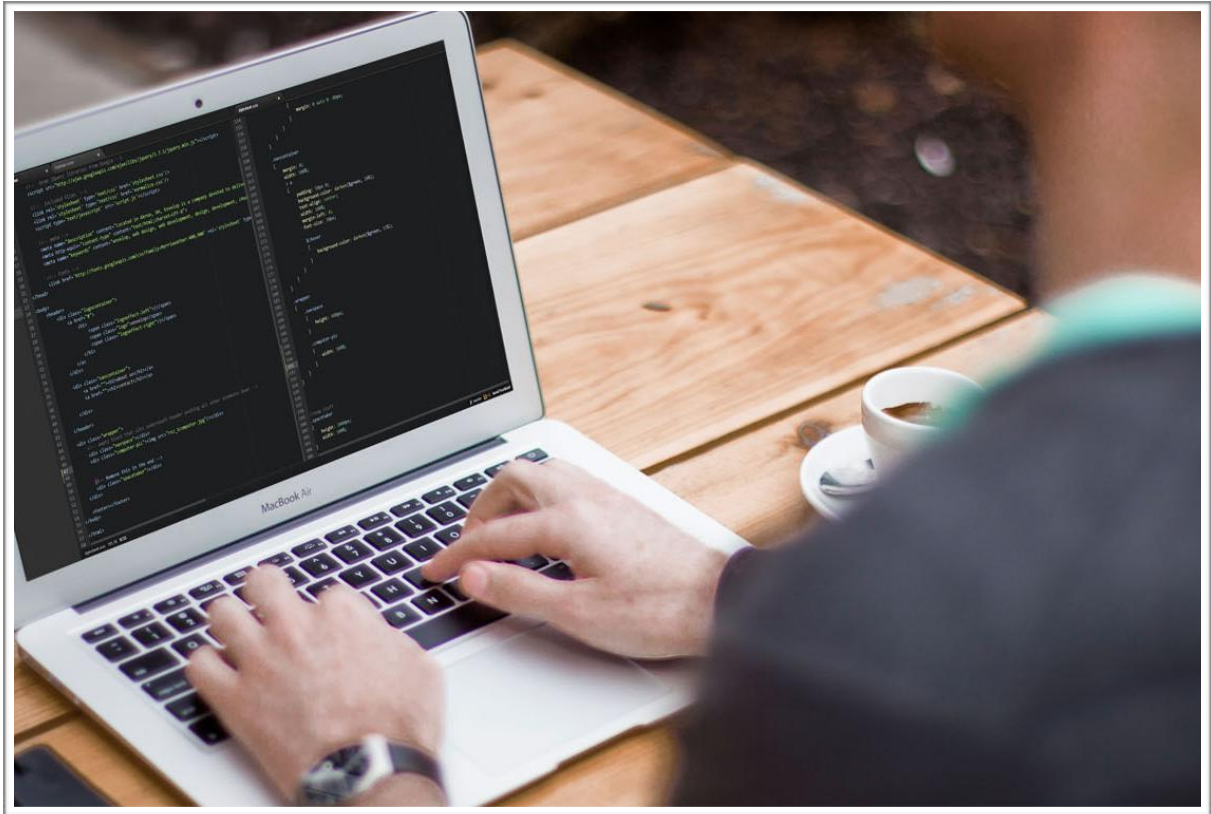


# Codenza C# Guide



Codenza Development Team

Fall 2017

# Codenza C# Guide



## Beginner User

MonoDevelop is Cross platform IDE for C#, F# and more and is based on the Open Source version of the .NET Framework.

Note: - Probably the better option if you do not want to install the Bulkier(though selectable component) Visual Studio IDE.

Download and install MonoDevelop from the link below: -

<http://www.mono-project.com/download/>

System requirements: -

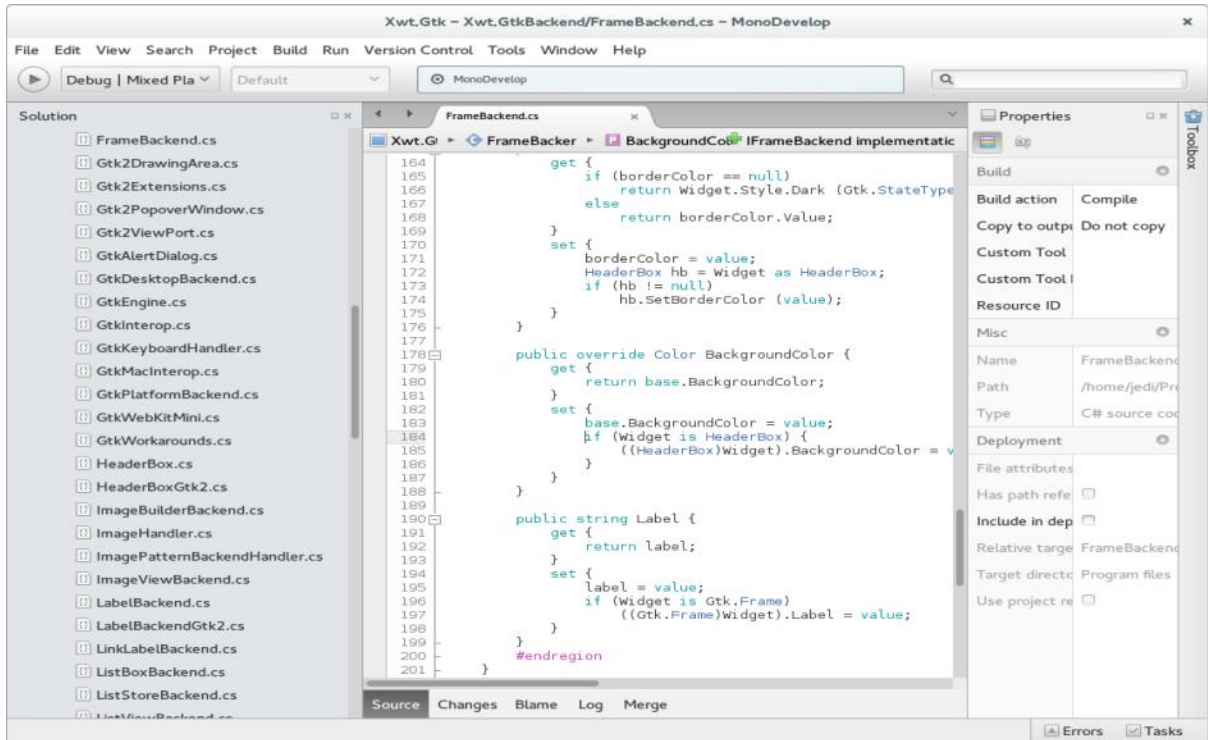
Windows – 32 / 64 bit.

Linux – 32 / 64 bit.

Mac OS - 32 / 64 bit.

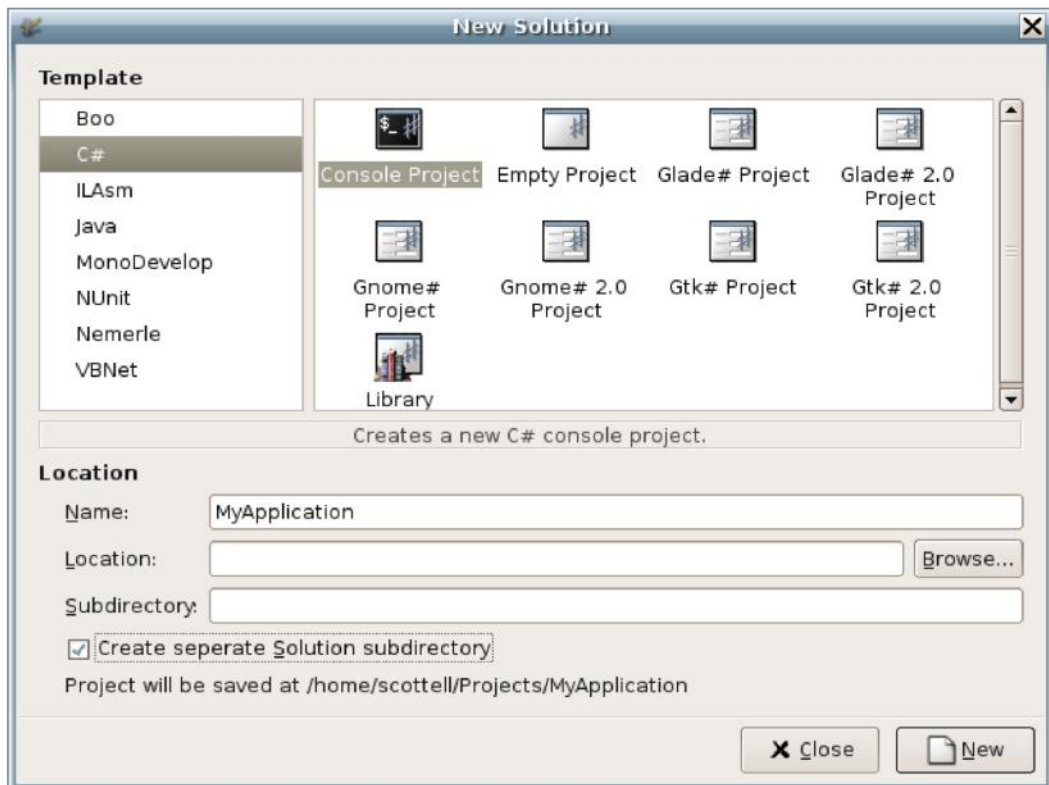
2GB RAM minimum / 4GB RAM recommended.

Disk Space: Upto 1GB (depending on features)



Building and executing C# Program:-

From the **File** menu, select “**New Project**”, this will open up the “New Project” window. Select “C#” from the language list and “**Console Project**” from the templates. Give your application a name as so:



When you are finished, click the “New” button. This creates a new directory for your solution in the Projects directory of your home directory. The “Console Project” template provides an already buildable application.

Eg: - Console Hello World

To test the most basic functionality available, copy the following code into a file called hello.cs

```
using System;

public class HelloWorld
{
    static public void Main ()
    {
        Console.WriteLine ("Hello Mono World");
    }
}
```

You can test by selecting “Run” from the “Run” menu. This will build the application, and provide the output in an “Application Output” tab.

Visual Studio Community Edition: -

It is a Fully-featured integrated development environment (IDE) for Android, iOS, Windows, web, and cloud. Though it is a paid software, the Community Edition is available with tons of features at free of cost.

Download and install from the link below: -

<https://www.visualstudio.com/vs/>

System Requirements: -

Windows – 7SP1, 8, 8.1, 10

Mac OS – as (Visual Studio for Mac)

1.6 GHz processor or better.

1GB RAM minimum / 4GB RAM minimum.

4GB HDD space or more (depending on features.)

Direct-X 9 capable graphics card (1024x768 min display resolution.)

Building and executing C# program: -

Create and run a console application

1. Start Visual Studio.
2. On the menu bar, choose **File, New, Project**.

The **New Project** dialog box opens.

3. Expand Installed, expand Templates, expand Visual C#, and then choose Console Application.

4. In the **Name** box, specify a name for your project, and then choose the **OK** button.

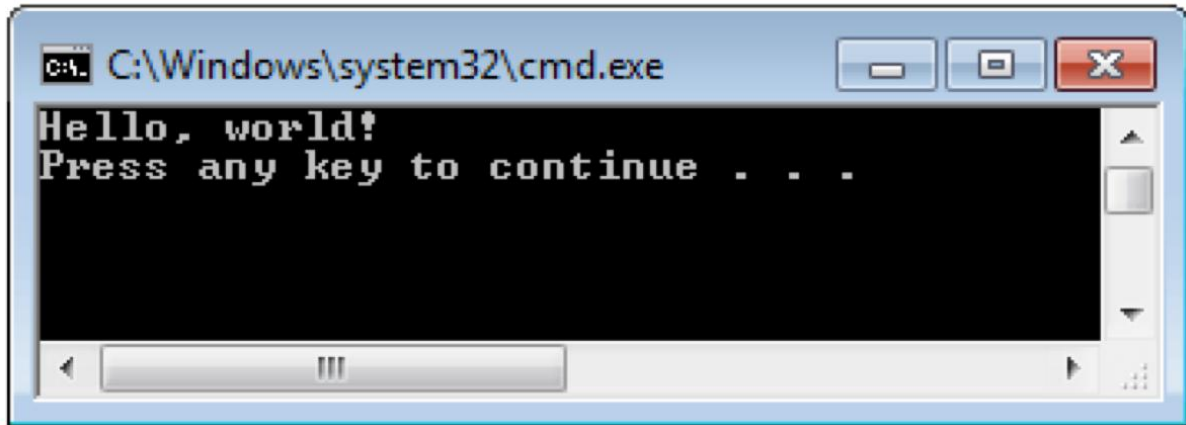
The new project appears in **Solution Explorer**.

5. If Program.cs isn't open in the **Code Editor**, open the shortcut menu for **Program.cs** in **Solution Explorer**, and then choose **View Code**.

6. Replace the contents of Program.cs with the following code.

```
C#  
  
// A Hello World! program in C#.  
using System;  
namespace HelloWorld  
{  
    class Hello  
    {  
        static void Main()  
        {  
            Console.WriteLine("Hello World!");  
  
            // Keep the console window open in debug mode.  
            Console.WriteLine("Press any key to exit.");  
            Console.ReadKey();  
        }  
    }  
}
```

7. Choose the F5 key to run the project. A Command Prompt window appears that contains the line Hello World!



CodeRunner IDE: -

An advanced, highly flexible, and easy-to-use **programming editor for your Mac**. CodeRunner supports a large number of languages, and delivers big IDE features while remaining lightweight and clutter-free.

System Requirements: -

Mac OS – 10.7+ (latest version preferred).

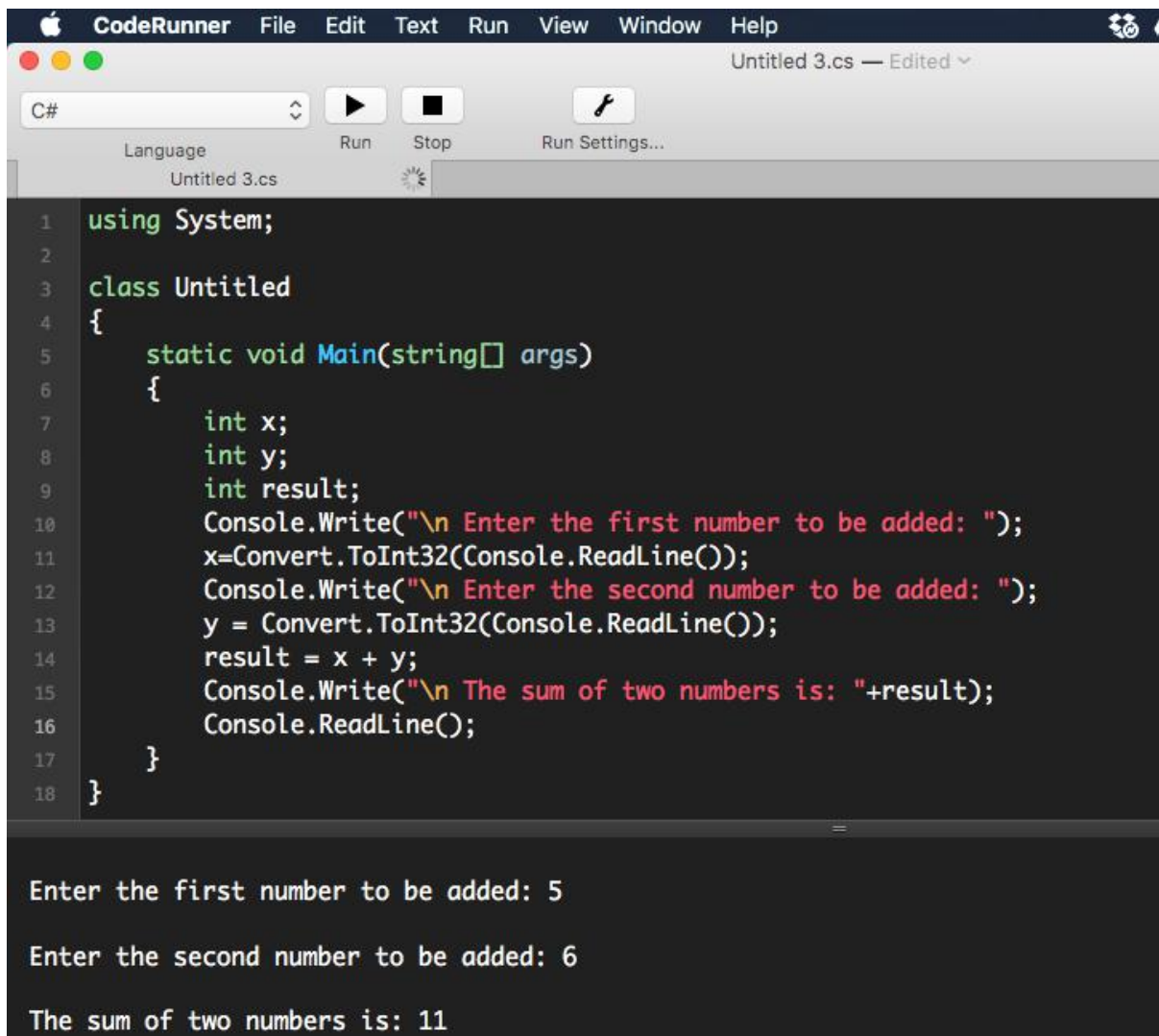
Xcode (latest version preferred).

Xcode Command Line Tools.

Note: - Though it is a paid software , it is also available as a Trial Software.

Building and executing the Program: -

Create New Program as **Click File -> New -> Select Programming Language (C#)**.



The screenshot shows the CodeRunner application window. The title bar includes 'CodeRunner', 'File', 'Edit', 'Text', 'Run', 'View', 'Window', and 'Help'. The current file is 'Untitled 3.cs'. The code editor contains the following C# code:

```
1 using System;
2
3 class Untitled
4 {
5     static void Main(string[] args)
6     {
7         int x;
8         int y;
9         int result;
10        Console.WriteLine("\n Enter the first number to be added: ");
11        x=Convert.ToInt32(Console.ReadLine());
12        Console.WriteLine("\n Enter the second number to be added: ");
13        y = Convert.ToInt32(Console.ReadLine());
14        result = x + y;
15        Console.WriteLine("\n The sum of two numbers is: "+result);
16        Console.ReadLine();
17    }
18 }
```

The output window below the code shows the following text:

```
Enter the first number to be added: 5
Enter the second number to be added: 6
The sum of two numbers is: 11
```

## Expert User

Expert users prefer the command line interface and assuming you have a C#(.cs) File.

Append the path of your C# Compiler to your system path and run C# Programs as **csc csharp\_file.cs**

