CHAPTER 2

■ ■ ■

Compile and Run

Visual Studio compilation

With the Hello World program completed, the next step is to compile and run it. To do so open up the Debug menu and select Start Without Debugging, or simply press Ctrl + F5. Visual Studio will then compile and run the application which displays the string in a console window.

The reason why you do not want to choose the Start Debugging command (F5) is because the console window will then close as soon as the program has finished executing.

Console compilation

If you did not have an IDE such as Visual Studio, you could still compile the program as long as you have the .NET Framework installed. To try this, open up a console window (C:\Windows\System32\cmd.exe) and navigate to the project folder where the source file is located. You then need to find the C# compiler called csc.exe, which is located in a path similar to the one shown below. Run the compiler with the source filename as an argument and it will produce an executable in the current folder.

```
C:\MySolution\MyProject>
\Windows\Microsoft.NET\Framework64\v2.0.50727\
csc.exe Program.cs
```

If you try running the compiled program it will show the same output as that created by Visual Studio.

```
C:\MySolution\MyProject> Program.exe
Hello World
```

Comments

Comments are used to insert notes into the source code. C# uses the standard C++ comment notations, with both single-line and multi-line comments. They are meant only to enhance the readability of the source code and have no effect on the end program. The single-line comment begins with "//" and extends to the end of the line. The multi-line comment may span multiple lines and is delimited by "/*" and "*/".

```
// single-line comment

/* multi-line
   comment */
```

In addition to these, there are two documentation comments. One single-line documentation comment that starts with "///", and one multi-line documentation comment that is delimited by "/**" and "*/". These comments are used when producing class documentation.

```
/// <summary>Class level documentation.</summary>
class MyApp
{
  /** <summary>Program entry point.</summary>
      <param name="args">Command line arguments.</param>
   */
  static void Main(string[] args)
  {
    System.Console.WriteLine("Hello World");
  }
}
```