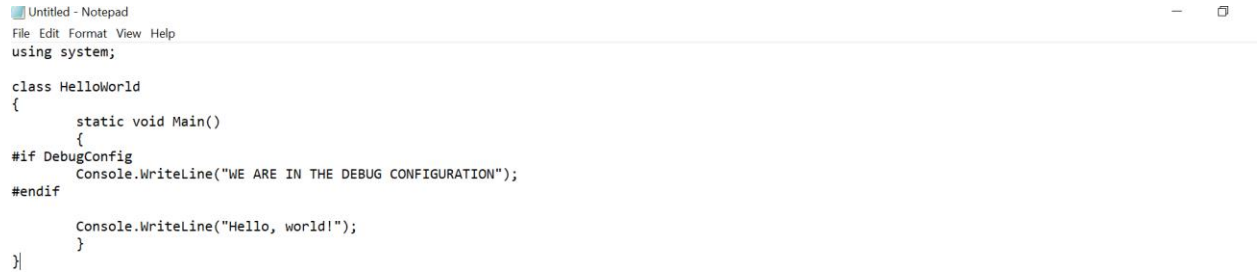


Creating an MSBuild project file from scratch

1. This shows how to create a minimal C# application source file by using a text editor.



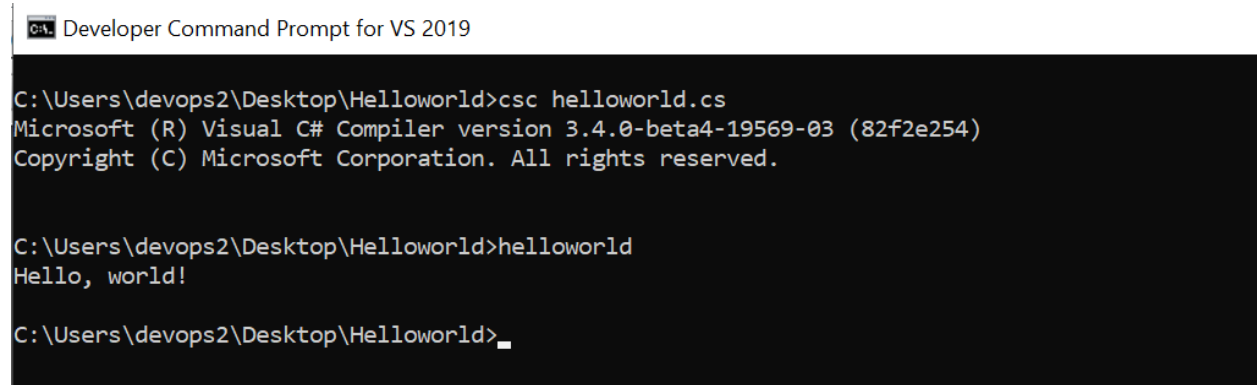
The screenshot shows a Notepad window titled "Untitled - Notepad". The menu bar includes "File", "Edit", "Format", "View", and "Help". The code is as follows:

```
using System;

class HelloWorld
{
    static void Main()
    {
        #if DebugConfig
        Console.WriteLine("WE ARE IN THE DEBUG CONFIGURATION");
        #endif

        Console.WriteLine("Hello, world!");
    }
}
```

2. Build the application by typing **csc helloworld.cs** at the command prompt.



The screenshot shows the "Developer Command Prompt for VS 2019" window. The command prompt shows the following sequence of commands and output:

```
C:\Users\devops2\Desktop\Helloworld>csc helloworld.cs
Microsoft (R) Visual C# Compiler version 3.4.0-beta4-19569-03 (82f2e254)
Copyright (C) Microsoft Corporation. All rights reserved.

C:\Users\devops2\Desktop\Helloworld>helloworld
Hello, world!

C:\Users\devops2\Desktop\Helloworld>_
```

3. Creating a minimal MSBuild project file

Untitled - Notepad

File Edit Format View Help

```
<Project xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <ItemGroup>
    <Compile Include="helloworld.cs" />
  </ItemGroup>

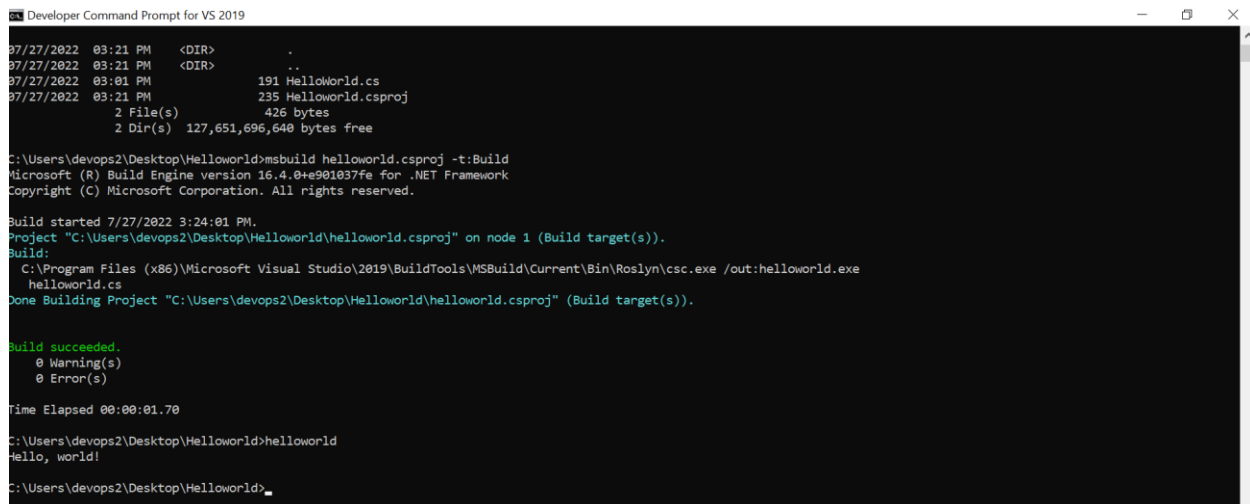
  <Target Name="Build">
    <Csc Sources="@ (Compile)" />
  </Target>

</Project>
```

4. At the command prompt, type **msbuild helloworld.csproj -t:Build**.

This builds the Build target of the Helloworld project file by invoking the Visual C# compiler to create the Helloworld application. Test the application by typing **helloworld**.

The **Hello, world!** message should be displayed.



```
Developer Command Prompt for VS 2019

07/27/2022 03:21 PM <DIR> .
07/27/2022 03:21 PM <DIR> ..
07/27/2022 03:01 PM      191 Helloworld.cs
07/27/2022 03:21 PM      235 Helloworld.csproj
                2 File(s)      426 bytes
                2 Dir(s) 127,651,696,640 bytes free

C:\Users\devops2\Desktop\Helloworld>msbuild helloworld.csproj -t:Build
Microsoft (R) Build Engine version 16.4.0+e901037fe for .NET Framework
Copyright (C) Microsoft Corporation. All rights reserved.

Build started 7/27/2022 3:24:01 PM.
Project "C:\Users\devops2\Desktop\Helloworld\helloworld.csproj" on node 1 (Build target(s)).
Build:
  C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin\Roslyn\csc.exe /out:helloworld.exe
  helloworld.cs
Done Building Project "C:\Users\devops2\Desktop\Helloworld\helloworld.csproj" (Build target(s)).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:01.70

C:\Users\devops2\Desktop\Helloworld>helloworld
Hello, world!

C:\Users\devops2\Desktop\Helloworld>
```

5. Adding build properties

➔ You can add build properties to the project file to further control the build. Now add these properties:

- An `AssemblyName` property to specify the name of the application.
- An `OutputPath` property to specify a folder to contain the application.

 Helloworld - Notepad

File Edit Format View Help

```
<Project xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <AssemblyName>MSBuildSample</AssemblyName>
    <OutputPath>Bin\</OutputPath>
  </PropertyGroup>
  <ItemGroup>
    <Compile Include="helloworld.cs" />
  </ItemGroup>

  <Target Name="Build">
    <MakeDir Directories="$(OutputPath)" Condition="!Exists('$(OutputPath)')"/>
    <Csc Sources="@ (Compile)" OutputAssembly="$(OutputPath)$(AssemblyName).exe" />
  </Target>
</Project>
```

6. Test the build properties

➔ Now you can build the application by using the project file in which you used build properties to specify the output folder and application name.

1. At the command prompt, type **msbuild helloworld.csproj -t:Build**.

This creates the `\Bin\` folder and then invokes the Visual C# compiler to create the *MSBuildSample* application and puts it in the `\Bin\` folder.

2. To verify that the `\Bin\` folder has been created, and that it contains the *MSBuildSample* application, type **dir Bin**.
3. Test the application by typing **Bin\MSBuildSample**.

The **Hello, world!** message should be displayed.

```
Developer Command Prompt for VS 2019
Done Building Project "C:\Users\devops2\Desktop\Helloworld\helloworld.csproj" (Build target(s)).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:01.68

C:\Users\devops2\Desktop\Helloworld>dir bin
Volume in drive C has no label.
Volume Serial Number is 0A02-8F3A

Directory of C:\Users\devops2\Desktop\Helloworld\bin

27/27/2022  03:34 PM  <DIR>          .
27/27/2022  03:34 PM  <DIR>          ..
27/27/2022  03:34 PM                3,584 MSBuildSample.exe
               1 File(s)                3,584 bytes
               2 Dir(s)  127,619,952,640 bytes free

C:\Users\devops2\Desktop\Helloworld>Bin\MSBuildSample
Hello, world!

C:\Users\devops2\Desktop\Helloworld>
```

7. Adding build targets

➔ Next, add two more targets to the project file, as follows:

- A Clean target that deletes old files.
- A Rebuild target that uses the DependsOnTargets attribute to force the Clean task to run before the Build task.

Now that you have multiple targets, you can set the Build target as the default target.

```
Helloworld - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8" standalone="no" ?>
<Project xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <AssemblyName>MSBuildSample</AssemblyName>
    <OutputPath>Bin</OutputPath>
  </PropertyGroup>
  <ItemGroup>
    <Compile Include="helloworld.cs" />
  </ItemGroup>
  <Target Name="Build">
    <MakeDir Directories="$(OutputPath)" Condition="!Exists('$(OutputPath)')"/>
    <Csc Sources="@$(Compile)" OutputAssembly="$(OutputPath)$(AssemblyName).exe" />
  </Target>
  <Target Name="Clean">
    <Delete Files="$(OutputPath)$(AssemblyName).exe" />
  </Target>
  <Target Name="Rebuild" DependsOnTargets="Clean;Build" />
</Project>
```

To test the build targets

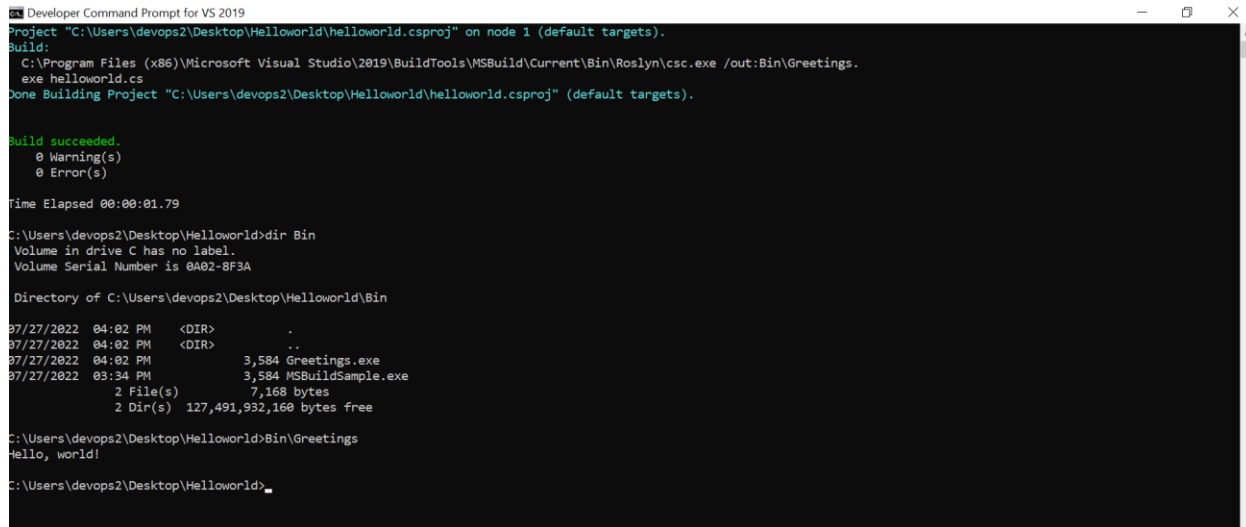
1. At the command prompt, type **msbuild helloworld.csproj -p:AssemblyName=Greetings**.

Because you did not use the **-t** switch to explicitly set the target, MSBuild runs the default Build target. The **-p** switch overrides the AssemblyName property and gives it

the new value, *Greetings*. This causes a new application, *Greetings.exe*, to be created in the `\Bin\` folder.

2. To verify that the `\Bin\` folder contains both the *MSBuildSample* application and the new *Greetings* application, type **dir Bin**.
3. Test the *Greetings* application by typing **Bin\Greetings**.

The **Hello, world!** message should be displayed.



```
Developer Command Prompt for VS 2019
Project "C:\Users\devops2\Desktop\Helloworld\helloworld.csproj" on node 1 (default targets).
Build:
  C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin\Roslyn\csc.exe /out:Bin\Greetings.exe helloworld.cs
Done Building Project "C:\Users\devops2\Desktop\Helloworld\helloworld.csproj" (default targets).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:01.79

C:\Users\devops2\Desktop\Helloworld>dir Bin
Volume in drive C has no label.
Volume Serial Number is 0A02-8F3A

Directory of C:\Users\devops2\Desktop\Helloworld\Bin

27/27/2022  04:02 PM  <DIR>          .
27/27/2022  04:02 PM  <DIR>          ..
27/27/2022  04:02 PM                3,584 Greetings.exe
27/27/2022  03:34 PM                3,584 MSBuildSample.exe
                2 File(s)              7,168 bytes
                2 Dir(s)  127,491,932,160 bytes free

C:\Users\devops2\Desktop\Helloworld\Bin\Greetings
Hello, world!

C:\Users\devops2\Desktop\Helloworld>
```

➔ Delete the *MSBuildSample* application by typing **msbuild helloworld.csproj -t:clean**.

This runs the Clean task to remove the application that has the default `AssemblyName` property value, *MSBuildSample*.

➔ Delete the *Greetings* application by typing **msbuild helloworld.csproj -t:clean -p:AssemblyName=Greetings**.

This runs the Clean task to remove the application that has the given **AssemblyName** property value, *Greetings*.

1. To verify that the `\Bin\` folder is now empty, type **dir Bin**.
2. Type **msbuild**.

Although a project file is not specified, MSBuild builds the *helloworld.csproj* file because there is only one project file in the current folder. This causes the *MSBuildSample* application to be created in the `\Bin\` folder.

```
Developer Command Prompt for VS 2019

C:\Users\devops2\Desktop\HelloWorld1>msbuild helloworld.csproj -t:Build
Microsoft (R) Build Engine version 16.4.0+g901037fe for .NET Framework
Copyright (C) Microsoft Corporation. All rights reserved.

Build started 7/28/2022 11:13:57 AM.
Project "C:\Users\devops2\Desktop\HelloWorld1\helloworld.csproj" on node 1 (Build target(s)).
Build:
  Creating directory "Bin\".
  C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin\Roslyn\csc.exe /out:Bin\MSBuildSample.exe helloworld.cs
Done Building Project "C:\Users\devops2\Desktop\HelloWorld1\helloworld.csproj" (Build target(s)).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:01.71

C:\Users\devops2\Desktop\HelloWorld1>dir Bin
Volume in drive C has no label.
Volume Serial Number is 0A02-8F3A

Directory of C:\Users\devops2\Desktop\HelloWorld1\Bin

27/28/2022  11:13 AM    <DIR>          .
27/28/2022  11:13 AM    <DIR>          ..
27/28/2022  11:13 AM                3,584 MSBuildSample.exe
               1 File(s)                3,584 bytes
               2 Dir(s)      124,850,151,424 bytes free

C:\Users\devops2\Desktop\HelloWorld1>cd Bin
C:\Users\devops2\Desktop\HelloWorld1\Bin>MSBuildSample
Hello, world!

C:\Users\devops2\Desktop\HelloWorld1\Bin>
```

8. Building incrementally

- ➔ You can tell MSBuild to build a target only if the source files or target files that the target depends on have changed. MSBuild uses the time stamp of a file to determine whether it has changed.

```
HelloWorld - Notepad
File Edit Format View Help

<Project DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <AssemblyName>MSBuildSample</AssemblyName>
    <OutputPath>Bin\</OutputPath>
  </PropertyGroup>
  <ItemGroup>
    <Compile Include="helloworld.cs" />
  </ItemGroup>
  <Target Name="Build" Inputs="@(<Compile>)" Outputs="$(OutputPath)$(AssemblyName).exe">
    <MakeDir Directories="$(OutputPath)" Condition="!Exists('$(OutputPath)') />
    <Csc Sources="@(<Compile>)" OutputAssembly="$(OutputPath)$(AssemblyName).exe" />
  </Target>

  <Target Name="Clean" >
    <Delete Files="$(OutputPath)$(AssemblyName).exe" />
  </Target>
  <Target Name="Rebuild" DependsOnTargets="Clean;Build" />
</Project>
```

Test the Build target by typing **msbuild -v:d** at the command prompt.

- ➔ Remember that *helloworld.csproj* is the default project file, and that Build is the default target.
- ➔ The **-v:d** switch specifies a verbose description for the build process.
- ➔ These lines should be displayed:

Skipping target "Build" because all output files are up-to-date with respect to the input files.

Input files: HelloWorld.cs

Output files: BinMSBuildSample.exe

- ➔ MSBuild skips the Build target because none of the source files have changed since the application was last built.

```
Developer Command Prompt for VS 2019
C:\Users\devops2\Desktop\HelloWorld1>msbuild -v:d
Microsoft (R) Build Engine version 16.4.0+9901837fe for .NET Framework
Copyright (C) Microsoft Corporation. All rights reserved.

Build started 7/28/2022 11:21:04 AM.
Project "C:\Users\devops2\Desktop\HelloWorld1\HelloWorld.csproj" on node 1 (default targets).
Building with tools version "Current".
Target "Build" in project "C:\Users\devops2\Desktop\HelloWorld1\HelloWorld.csproj" (entry point):
Building target "Build" completely.
Output file "Bin\MSBuildSample.exe" does not exist.
Task "MakeDir" skipped, due to false condition; (!Exists('${OutputPath}')) was evaluated as (!Exists('Bin\')).
Using "Csc" task from assembly "C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin\Roslyn\Microsoft.Build.Tasks.CodeAnalysis.dll".
Task "Csc"
  C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Current\Bin\Roslyn\csc.exe /out:Bin\MSBuildSample.exe helloworld.cs
  Microsoft (R) Visual C# Compiler version 3.4.0-beta4-19569-03 (82f2e254)
  Copyright (C) Microsoft Corporation. All rights reserved.

Done executing task "Csc".
Done building target "Build" in project "HelloWorld.csproj".
Done Building Project "C:\Users\devops2\Desktop\HelloWorld1\HelloWorld.csproj" (default targets).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:01.79

C:\Users\devops2\Desktop\HelloWorld1>
```