C# language versioning

Article • 11/30/2022

The latest C# compiler determines a default language version based on your project's target framework or frameworks. Visual Studio doesn't provide a UI to change the value, but you can change it by editing the *csproj* file. The choice of default ensures that you use the latest language version compatible with your target framework. You benefit from access to the latest language features compatible with your project's target. This default choice also ensures you don't use a language that requires types or runtime behavior not available in your target framework. Choosing a language version newer than the default can cause hard to diagnose compile-time and runtime errors.

C# 11 is supported only on .NET 7 and newer versions. C# 10 is supported only on .NET 6 and newer versions. C# 9 is supported only on .NET 5 and newer versions.

Check the Visual Studio platform compatibility page for details on which .NET versions are supported by versions of Visual Studio. Check the Visual Studio for Mac platform compatibility page for details on which .NET versions are supported by versions of Visual Studio for Mac. Check the Mono page for C# ⬀ for Mono compatibility with C# versions.

Defaults

The compiler determines a default based on these rules:

Target	Version	C# language version default
.NET	7.x	C# 11
.NET	6.x	C# 10
.NET	5.x	C# 9.0
.NET Core	3.x	C# 8.0
.NET Core	2.x	C# 7.3
.NET Standard	2.1	C# 8.0
.NET Standard	2.0	C# 7.3
.NET Standard	1.x	C# 7.3
.NET Framework	all	C# 7.3

When your project targets a preview framework that has a corresponding preview language version, the language version used is the preview language version. You use the latest features with that preview in any environment, without affecting projects that target a released .NET Core version.

> ⓘ **Important**
>
> The new project template for Visual Studio 2017 added a `<LangVersion>latest</LangVersion>` entry to new project files. If you upgrade the target framework for these projects, they **override the default behavior**. You should remove the `<LangVersion>latest</LangVersion>` from your project file when you update the .NET SDK. Then, your project will use the compiler version recommended for your target framework. You can update the target framework to access newer language features.

Override a default

If you must specify your C# version explicitly, you can do so in several ways:

- Manually edit your [project file](#).
- Set the language version [for multiple projects in a subdirectory](#).
- Configure the **LangVersion** [compiler option](#).

> ♡ **Tip**
>
> You can see the language version in Visual Studio in the project properties page. Under the *Build* tab, the *Advanced* pane displays the version selected.
>
> To know what language version you're currently using, put `#error version` (case sensitive) in your code. This makes the compiler report a compiler error, CS8304, with a message containing the compiler version being used and the current selected language version. See **#error (C# Reference)** for more information.

Edit the project file

You can set the language version in your project file. For example, if you explicitly want access to preview features, add an element like this:

XML

```xml
<PropertyGroup>
    <LangVersion>preview</LangVersion>
</PropertyGroup>
```

The value `preview` uses the latest available preview C# language version that your compiler supports.

Configure multiple projects

To configure multiple projects, you can create a **Directory.Build.props** file that contains the `<LangVersion>` element. You typically do that in your solution directory. Add the following to a **Directory.Build.props** file in your solution directory:

```xml
XML

<Project>
 <PropertyGroup>
    <LangVersion>preview</LangVersion>
 </PropertyGroup>
</Project>
```

Builds in all subdirectories of the directory containing that file will use the preview C# version. For more information, see Customize your build.

C# language version reference

The following table shows all current C# language versions. Your compiler may not necessarily understand every value if it's older. If you install the latest .NET SDK, then you have access to everything listed.

Value	Meaning
preview	The compiler accepts all valid language syntax from the latest preview version.
latest	The compiler accepts syntax from the latest released version of the compiler (including minor version).
latestMajor or default	The compiler accepts syntax from the latest released major version of the compiler.
11.0	The compiler accepts only syntax that is included in C# 11 or lower.
10.0	The compiler accepts only syntax that is included in C# 10 or lower.
9.0	The compiler accepts only syntax that is included in C# 9 or lower.

Value	Meaning
`8.0`	The compiler accepts only syntax that is included in C# 8.0 or lower.
`7.3`	The compiler accepts only syntax that is included in C# 7.3 or lower.
`7.2`	The compiler accepts only syntax that is included in C# 7.2 or lower.
`7.1`	The compiler accepts only syntax that is included in C# 7.1 or lower.
`7`	The compiler accepts only syntax that is included in C# 7.0 or lower.
`6`	The compiler accepts only syntax that is included in C# 6.0 or lower.
`5`	The compiler accepts only syntax that is included in C# 5.0 or lower.
`4`	The compiler accepts only syntax that is included in C# 4.0 or lower.
`3`	The compiler accepts only syntax that is included in C# 3.0 or lower.
`ISO-2` or `2`	The compiler accepts only syntax that is included in ISO/IEC 23270:2006 C# (2.0).
`ISO-1` or `1`	The compiler accepts only syntax that is included in ISO/IEC 23270:2003 C# (1.0/1.2).