Which version of C# am I using

Asked 5 years, 11 months ago   Active 13 days ago   Viewed 112k times

▲

159

▼

★

30

My question is as easy as it seems: I want to find out which version of C# I'm using. If I would be using python I would do something like `python -v` from command line, or type

```
import sys
print sys.version
```

In php I would do something like this: `phpinfo();` in java: `java -version`

But I was not able to find how to achieve this in C#.

I'm completely new to C#, so please do not beat me too hard. What surprises me that I was not able to find this answer on SO, therefore I think that I'm missing something or bad in finding things. BTW, this question does not answer it, although the name suggests that it should.

Thank you for answers. Right now I got that it depends on .NET framework, but is there a programmatic way of figuring out my framework? (without going to the directory and checking the name of my .NET folders)

c#   version

edited Jul 18 at 11:10   asked Oct 23 '13 at 4:42

RBT   Salvador Dali
11.1k 7   91   115   **130k** 92   542   642

---

5   This is John Skeet's top rated question – Avi Turner Oct 23 '13 at 4:49

---

4   in Visual Studio: Project Properties > Build > Advanced... > Language Version: codeproject.com/Tips/865579/… – Slai Jul 24 '17 at 13:34

To get version of framework - look at version of one of main Assemblies i.e.

49

```
Console.Write(typeof(string).Assembly.ImageRuntimeVersion);
```

Getting version of C# compiler is somewhat harder, but you should be able to guess version by checking what framework version is used.

If you are using command line compiler (csc.exe) you can check help to see version (also you'd need to know Framework version anyway:

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319>csc /?
Microsoft (R) Visual C# 2010 Compiler version 4.0.30319.1
```

edited Jan 3 '17 at 16:02   answered Oct 23 '13 at 5:23

      Facundo La Rocca   **Alexei Levenkov**
      2,467   14   31   **87.9k**   9   96   143

---

18   Unfortunately Assembly.ImageRuntimeVersion does not tell you the right version number - on my PC, **.NET 4.6 RC** is installed but Assembly.ImageRuntimeVersion reports **v4.0.30319** – Matt Jul 14 '15 at 10:55

1   Unless you're using C# 6 which doesn't look at the .NET version. – Michael Puckett II Jan 27 '17 at 5:38

2   RE: Matt's comment: Just because you have one version of .NET installed on your PC, it doesn't mean that's the version in use in your project – LoJo Oct 26 '17 at 20:20

   @LoJo - You are absolutely right. One reason is that Visual Studio allows multi-targeting (i.e. you can select the .NET Target Framework version in the project properties), the other reason is that newer Roslyn versions need to be added via NUGET to your project. Third, as I wrote here, csc.exe is limited to version 5.0 and cannot be used to detect higher versions of C#. – Matt Jan 16 '18 at 14:30 ✎

2   Nobody has yet mentioned that the correct instruction to get the list of versions supported by your Roslyn compiler is "C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\MSBuild\15.0\Bin\Roslyn\csc.exe" /langversion:? – David A. Gray Aug 14 '18 at 1:25

It depends upon the .NET Framework that you use. Check **Jon Skeet's answer** about Versions.

Here is short version of his answer.

C# 1.2 (bizarrely enough); released with *.NET 1.1*

C# 2.0 released with *.NET 2.0*

C# 3.0 released with *.NET 3.5*

C# 4.0 released with *.NET 4*

C# 5.0 released with *.NET 4.5*

C# 6.0 released with *.NET 4.6*

C# 7.0 is released with *.NET 4.6.2*

C# 7.3 is released with *.NET 4.7.2*

edited Sep 23 '18 at 21:20	answered Oct 23 '13 at 4:46
Poul Bak	Praveen
5,577 3 12 33	**43.5k** 25 113 143

---

13 ... and **C# 6.0** is released with **.NET 4.6** – Matt Jul 14 '15 at 10:54 ✎

3 Just a heads up; C# 6 uses a different compiler (Roslyn) and will compile source code to other .NET versions not just .NET 4.6. – Michael Puckett II Jan 27 '17 at 5:38

6 ... and **C# 7.0** is released with **.NET 4.6.2** – Jivan Apr 4 '17 at 19:52 ✎

9 But you can compile .NET 2.0 with C# 7.0 compiler. Checking the .NET version is a very limited deduction and gives only a probable answer. – Abel Sep 25 '17 at 15:37 ✎

5 Let me add to @Abel's comment that this link shows you how you can change the C# language version (almost) independently from the framework. So you can only deduct from syntax errors you're getting that the code assumes a newer version of C# than you might have selected in the properties. – Matt Jan 18 '18 at 9:07 ✎
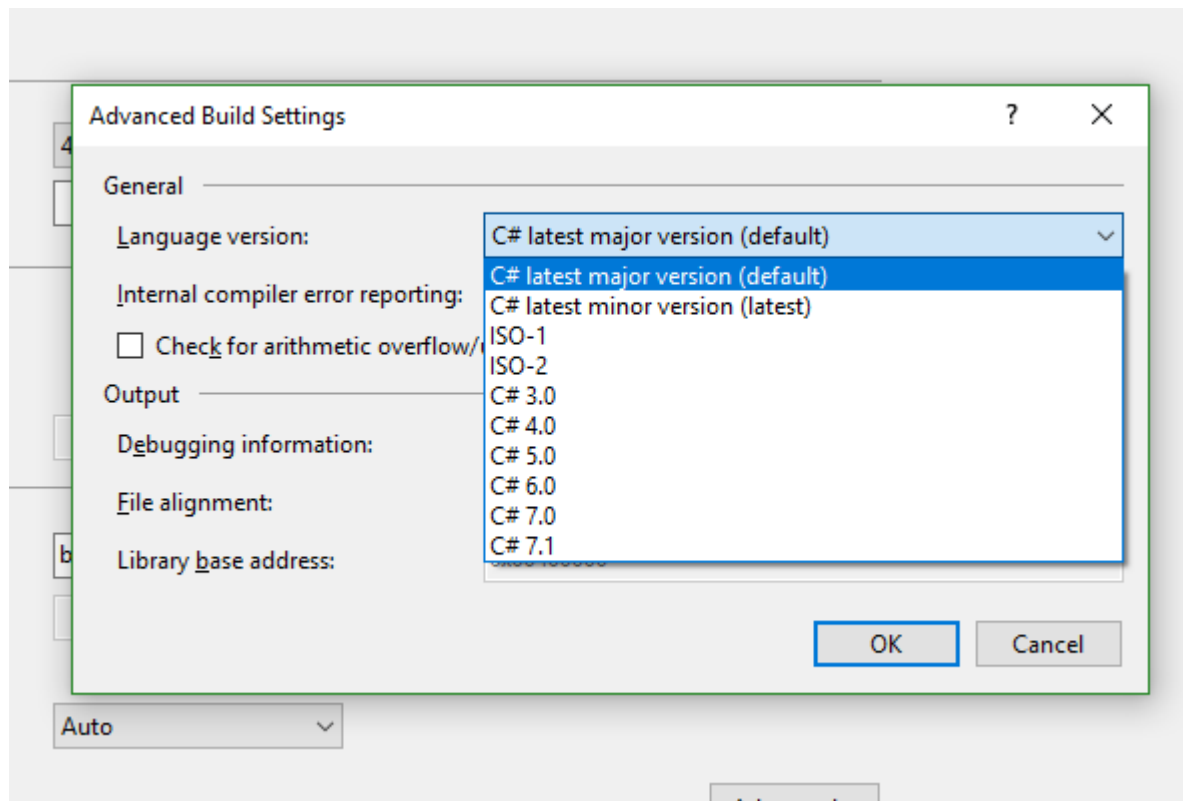

▲

77

While this isn't answering your question directly, I'm putting this here as google brought this page up first in my searches when I was looking for this info.

If you're using Visual Studio, you can right click on your project, go to project properties and see the version your project is using by

Advanced Build Settings   ?   X

General

Language version: C# latest major version (default)

                 C# latest major version (default)

Internal compiler error reporting: C# latest minor version (latest)

                 ISO-1

☐ Check for arithmetic overflow/   ISO-2

Output   C# 3.0

                 C# 4.0

Debugging information: C# 5.0

                 C# 6.0

File alignment: C# 7.0

                 C# 7.1

Library base address:

                                OK   Cancel

Auto

edited Nov 14 '17 at 4:55	answered Nov 3 '17 at 9:29
Rizan Zaky	Darth
1,147 2 10 28	**789** 4 6

The C# version you are using totally depends upon the .Net version you are using.

8

if you are using visual studio for development, you get to choose the .net framework version the c# version associated with it comes accordingly

These are the versions of C# known:

- **C# 1.0** released with **.NET 1.0** and VS2002 (January 2002)

- **C# 2.0** released with **.NET 2.0** and VS2005 (November 2005). Major new features: generics, anonymous methods, nullable types, iterator blocks

- **C# 3.0** released with **.NET 3.5** and VS2008 (November 2007). Major new features: lambda expressions, extension methods, expression trees, anonymous types, implicit typing (`var`), query expressions

- **C# 4.0** released with **.NET 4** and VS2010 (April 2010). Major new features: late binding (`dynamic`), delegate and interface generic variance, more COM support, named arguments and optional parameters

- **C# 5.0** released with **.NET 4.5** in August 2012.

Refrence **Jon Skeet's C# Versions Answer**

edited Aug 29 '17 at 4:47   answered Oct 23 '13 at 4:47

Vinay Pratap Singh
8,038   2   20   48

U can use some features from C# 3.0 in .NET 2.0 so it is not really 1-1 mapping. – nefarel Nov 4 '14 at 14:35

---

11   This is wrong - it doesn't depend on the .NET version, it depends on the compiler. – markmnl May 19 '16 at 0:48

---

this is plain wrong. My current project is set to .NET 4.0 and yet I'm still able to use C# 7.0 features. But once I push to remote then the CI/CD system failed to build because it uses C# version 5 or 6 – phuclv May 8 at 1:29

▲

6   **.NET version through registry**

▼   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\NET Framework Setup\NDP\ explore the children and look into each version. The one with key 'Full' is the version on the system.

https://support.microsoft.com/en-us/kb/318785 https://msdn.microsoft.com/en-us/library/hh925568(v=vs.110).aspx

.NET version through Visual Studio

Help -> About Microsoft Visual Studio -> The .NET version is specified on the top right.

As I understand at this time the Visual studio uses .NET Framework from the OS.

Through the dll

If you know the .NET Framework directory e.g. C:\Windows\Microsoft.NET\Framework64\v4.0.30319

Open System.dll, right click -> properties -> Details tab

C# version

Help -> About Microsoft Visual Studio

In the installed products lists there is Visual C#. In my case Visual C# 2015

Visual Studio (Microsoft) ships C# by name Visual C#.

https://msdn.microsoft.com/en-us/library/hh156499.aspx

C# 6, Visual Studio .NET 2015 Current version, see below

answered Dec 20 '16 at 13:42

Amod Pandey
635 2 9 20

By default following are corresponding version of C# compilers for Visual Studio:

5

1. Visual Studio 2015: C# 6.0

2. Visual Studio 2013: C# 5.0

3. Visual Studio 2012: C# 5.0

4. Visual Studio 2010: C# 4.0

5. Visual Studio 2008: C# 3.0

6. Visual Studio 2005: C# 2.0

7. Visual Studio.NET 2003: C# 1.2

8. Visual Studio.NET 2002: C# 1.0

Open the project properties window:

```
step 1. Right click on the Project Name
step 2. Select "Properties" (last option in menu)
step 3. Select "Build" from left hand side options and scroll till down
step 4. click on "Advance" button.
step 5. It will open a popup and there you will get "Language Version" dropdown
step 6. Select desired version of C# and Click "OK"
```

answered Apr 4 '18 at 9:12

Ankit Mori
365 4 14

3

In order to see the installed compiler version of VC#:

Open Visual Studio command prompt and just type csc then press Enter.

You will see something like following:

> Microsoft (R) Visual C# Compiler version 4.0.30319.34209
>
> for Microsoft (R) .NET Framework 4.5
>
> Copyright (C) Microsoft Corporation. All rights reserved.

P.S.: "**CSC**" stand for "**C Sharp Compiler**". Actually using this command you run csc.exe which is an executable file which is located in "c:\Windows\Microsoft.NET\Framework\vX.X.XXX". For more information about CSC visit http://www.file.net/process/csc.exe.html

edited Aug 3 '17 at 6:10

AlexRebula
536 2 7 20

answered Mar 17 '16 at 22:47

Siyavash Hamdi
2,153 1 11 24

For Windows, you run dev at the command/ search program line and select Developer Command Prompt for VS. Then you are going to

```
csc
```

Now you get information similar to

```
Microsoft (R) Visual C# Compiler version 2.6.0.62329 (5429b35d)
Copyright (C) Microsoft Corporation. All rights reserved.
```

For Windows and if you start with cmd terminal

```
cd C:\Windows\Microsoft.NET\Framework\
dir
```

Now you see all directories and files in .NET\Framework\ Please, select v... latest and go there, for example,

```
cd v4.0.30319
```

Run

```
csc
```

You will see information about version of C# compiler, that can be something similar to

```
Microsoft (R) Visual C# Compiler version 4.7.2556.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.
```

edited Apr 28 '18 at 16:30   answered Apr 28 '18 at 16:22

Roman
4,602   1   32   39

From developer command prompt type

That will display all C# versions supported including the default:

```
1
2
3
4
5
6
7.0 (default)
7.1
7.2
7.3 (latest)
```

edited May 8 at 1:33   answered Jan 28 at 16:02

phuclv   fortesl
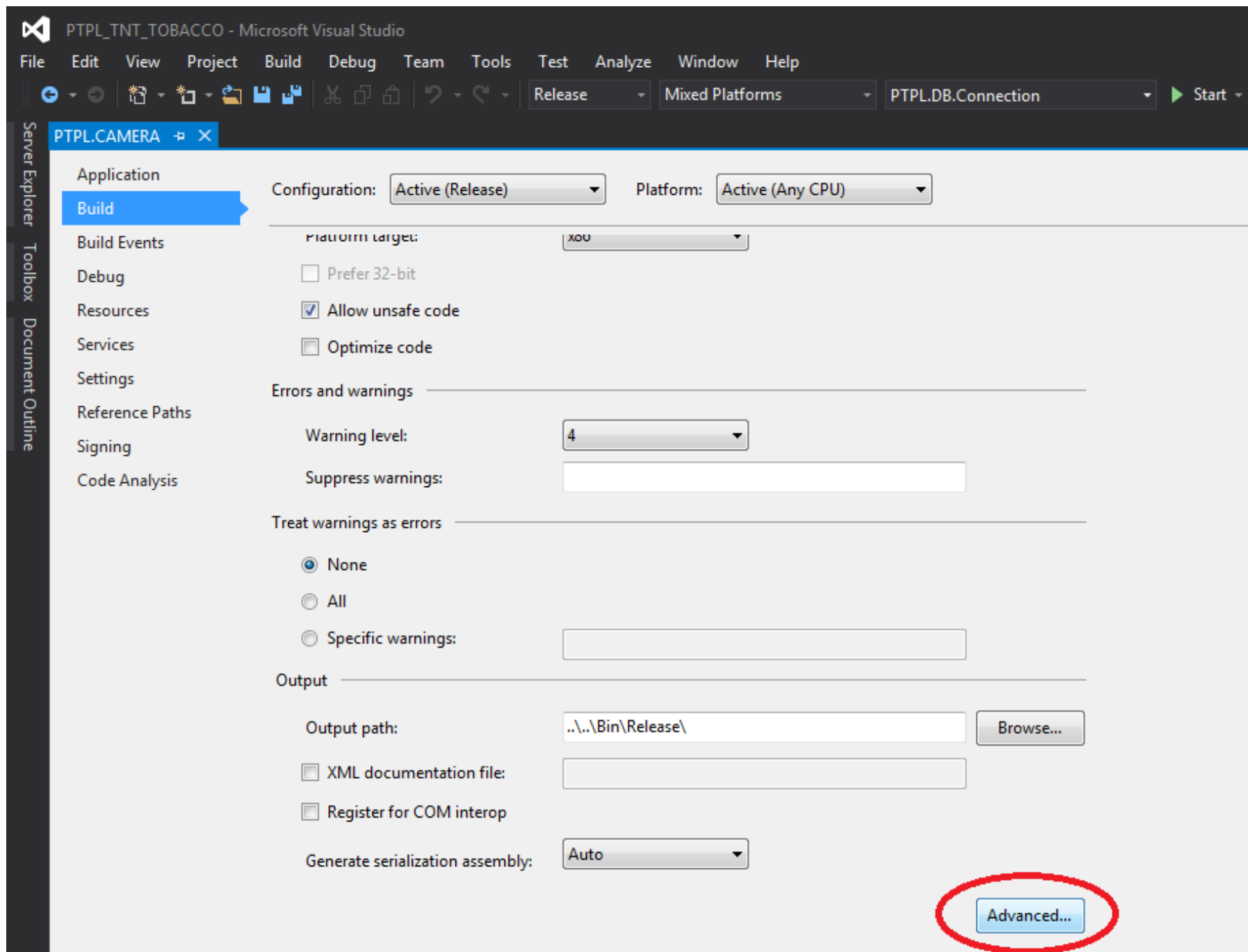**18.2k**   9   62   267   **156**   11

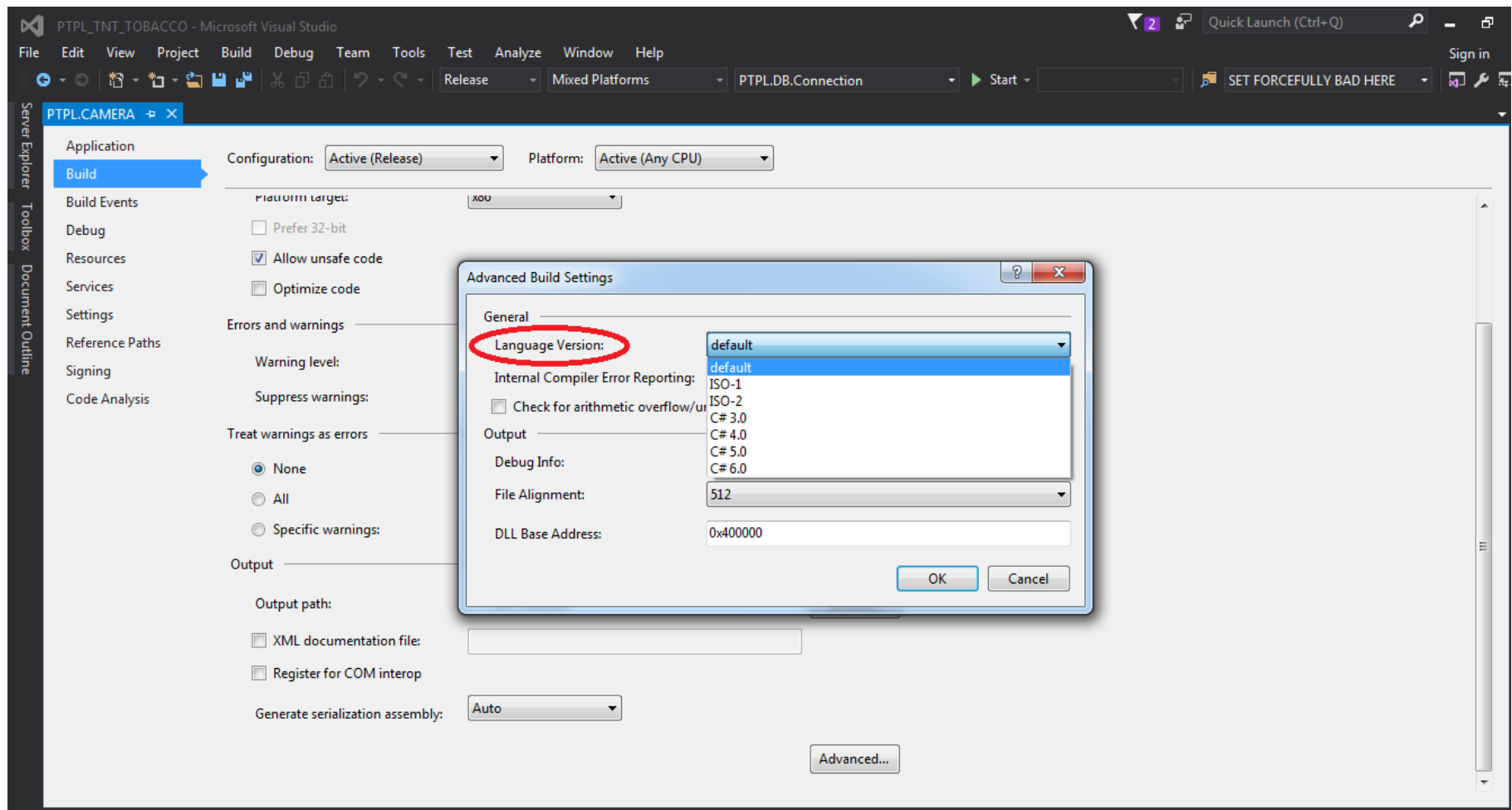If you are using VS2015 then follow below steps to find out the same:

2

1. Right click on the project.

2. Click on the *Properties* tab.

3. From properties window select *Build* option.

4. In that click on the *Advance* button.

5. There you will find out the language version.

Below images show the steps for the same:

Step 1:

phuclv   Sunil Jadhav
18.2k   9   62   267   **19**   3

Thanks to @fortesl and this answer

0   I'm using VS 2019 and it doesn't easily tell you the C# version you are using. I'm working with .Net Core 3.0, and VS 2019 using C# 8 in that environment. But "csc -langversion:?" makes this clear:

```
D:\>csc -langversion:?
Supported language versions:
default
1
2
3
4
5
6
7.0
7.1
7.2
7.3
8.0 (default)
latestmajor
preview
latest
```

Not sure what csc -version is identifying:

```
D:\>csc --version
Microsoft (R) Visual C# Compiler version 3.4.0-beta1-19462-14 (2f21c63b)
Copyright (C) Microsoft Corporation. All rights reserved.
```

answered Sep 24 at 7:52

guyr
11 1

Looks like OP needs a coding related answer – bradbury9 Sep 24 at 13:22