

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

Take the 2-minute tour



Difference between Build Solution, Rebuild Solution, and Clean Solution in Visual Studio?

CAREERS 2.0
by stackoverflow



+



Put all that writing to good use!
Update your profile today!

What is the difference between Build Solution, Rebuild Solution, and Clean Solution in Visual Studio?

When is the appropriate time to use each one of these?

visual-studio

asked Jun 22 '10 at 18:17



Fawa

1,491 2 7 3

3 Also see stackoverflow.com/questions/1247457/... – Greg Jun 22 '10 at 18:29

1 [Here is a nice post with details](#) – Zaheer Ahmed Nov 21 '13 at 8:36

add a comment

8 Answers

- **Build solution** will perform an incremental build: if it doesn't *think* it needs to rebuild a project, it won't. It may also use partially-built bits of the project if they haven't changed (I don't know how far it takes this)
- **Rebuild solution** will clean and then build the solution from scratch, ignoring anything it's done before
- **Clean solution** will remove the build artefacts from the previous build. If there are any other files in the build target directories (bin and obj) they may not be removed, but actual build artefacts are. I've seen behaviour for this vary - sometimes deleting fairly thoroughly and sometimes not - but I'll give VS the benefit of the doubt for the moment :)

(The links are to the devenv.exe command line switches, but they do the same as the menu items.)

edited Sep 28 '12 at 18:04



Tod

3,588 1 22 55

answered Jun 22 '10 at 18:21



Jon Skeet

710k 296 4622 5986

1 @womp: Not in the project I've just been looking at. It's still got all the assemblies there... – Jon Skeet Jun 22 '10 at 18:25

29 I have personally found "Clean Solution" to be more than unhelpful. If I *really* want it clean, manually deleting the bin and obj folders is the way to go. Have even been caught chasing phantom "errors" - until I did that. Clean just isn't reliable. – Chris Rogers May 26 '11 at 23:26

1 if artifacts made their way through other means than through build artifacts (say for example a copy from another source integrated as a msbuildtask in the cproj) then clean leaves them be. Which makes it pretty much useless, I'd even say it is dangerous as it will leave you with a false sense of cleanliness. – Newtopian Aug 30 '12 at 21:31

17 The link provided shows that Rebuild is a clean followed by a build, wouldn't that be helpful to mention? I don't have the hubris to edit Mr. Skeet's post. – Tod Sep 28 '12 at 18:00

3 @Tod: Please go ahead :) – Jon Skeet Sep 28 '12 at 18:01

show 7 more comments



Taken from [this link](#):

Build means compile and link only the source files that have changed since the last build, while Rebuild means compile and link all source files regardless of whether they changed or not. Build is the normal thing to do and is faster. Sometimes the versions of project target components can get out of sync and rebuild is necessary to make the build successful. In practice, you never need to Clean.

answered Jun 22 '10 at 18:19



[Matthew Jones](#)

14.1k 5 43 93

12 +1 because you were first. – [ChrisF](#) ♦ Jun 22 '10 at 18:23

1 [Here is a nice post with details](#) – [Zaheer Ahmed](#) Nov 21 '13 at 8:39

1 Link? I thought the idea of a DLL, aka Dynamic Link Library, was to link at runtime? – [The Dag](#) Mar 21 at 7:46

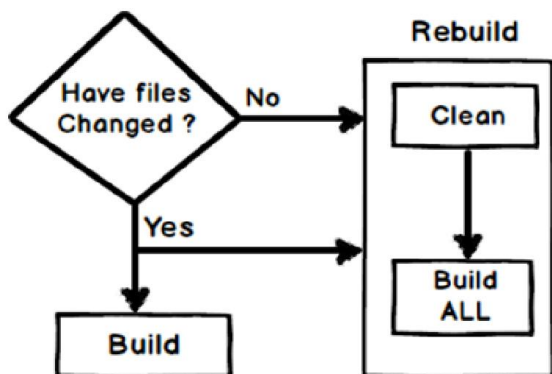
[add a comment](#)

Build solution :- Compiles code files (DLL and Exe) which are changed.

Rebuild :- Deletes all compiled file and Compiles them again irrespective if the code has changed or not.

Clean solution :- Deletes all compiled files (DLL and Exe).

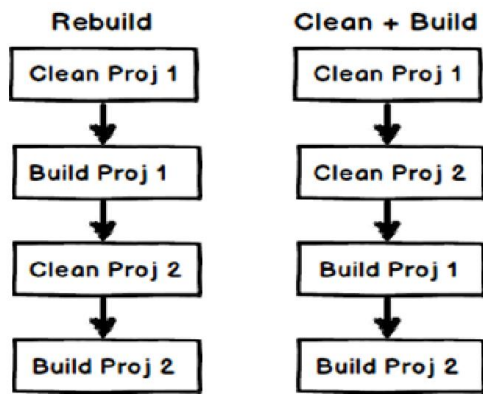
You can see this youtube video (<http://www.youtube.com/watch?v=4UIGXNbfWr8>) where i have demonstrated the differences and below are visual representation which will help you to analyze the same in more detail.



Just adding more to this answer the difference between Rebuild vs (Clean + Build) because there seems to be some confusion around this as well.

The difference is the way the build and clean sequence happens for every project. Let's say if your solution has two projects "proj1" and "proj2". If you do a rebuild it will take "proj1", clean (delete) the compiled files for "proj1" and build it. After that it will take the second project "proj2", clean compiled files for "proj2" and compile "proj2".

But if you do a "clean" and build". It will first delete all compiled files for "proj1" and "proj2" and then it will build "proj1" first followed by "proj2".



answered Jul 22 '13 at 6:24

 [Shivprasad Koirala](#)
1,689 13 16

3 Thank you for explaining why Rebuild doesn't always work and I often need to clean before doing a build. – [didibus](#) Jan 21 at 19:11

Nice explanation Siv Sir :) – [Suvendu Shekhar Giri](#) Jul 24 at 5:04

[add a comment](#)

Build Solution - Builds any assemblies which have changed files. If an assembly has no changes, it won't be re-built. Also will not delete any intermediate files.

Used most commonly.

Rebuild Solution - Rebuilds all assemblies regardless of changes but leaves intermediate files.

Used when you notice that Visual Studio didn't incorporate your changes in the latest assembly. Sometimes Visual Studio does make mistakes.

Clean Solution - Delete all intermediate files ~~and rebuild all assemblies regardless of changes~~

Used when all else fails and you need to clean everything up and start fresh.

answered Jun 22 '10 at 18:20

 [Justin Niessner](#)
146k 10 233 353

7 Clean doesn't do a build. – [Jon Skeet](#) Jun 22 '10 at 18:22

1 @Jon Skeet - Learn something new every day. I would've sworn it did a rebuild. I guess my memory isn't always as reliable as I'd like. – [Justin Niessner](#) Jun 22 '10 at 18:25

[add a comment](#)

I just think of Rebuild as performing the Clean first followed by the Build. Perhaps I am wrong ... comments?

answered Jun 22 '10 at 18:25

 [Ken Richards](#)
906 5 12

This had no upvotes (until now) and according to the docs (see the link in Jon's answer) this is exactly right. – [Tod](#) Sep 28 '12 at 17:57

1 I don't think it does. I have a situation where doing a Clean Solution, followed by Build Solution works, but doing a Rebuild Solution fails. This is on a freshly created solution with 2 projects (one a dependent of the other). – [Cthutu](#) May 28 '13 at 19:56

@Cthutu See Shivprasad's answer for the detail that makes the difference here. Rebuild cleans and then builds each individual project at a time, whereas running Clean first cleans everything out at once, then Build builds it all at once. I've run across instances where this change in clean/build order makes the difference between compiling and not compiling, too. – [Sean](#) Oct 15 '13 at 19:29

[add a comment](#)

Build solution will build any projects in the solution that have changed. Rebuild builds all projects no matter what, clean solution removes all temporary files ensuring that the next build is complete.

answered Jun 22 '10 at 18:20



SnOrfus

6,578 4 47 89

[add a comment](#)

Build solution only builds those projects which have changed in the solution, and does not effect assemblies that have not changed,

ReBuild first cleans, all the assemblies from the solution and then builds entire solution regardless of changes done.

Clean, simply cleans the solution.

answered Jun 19 at 17:31



yashashwi

12 3

[add a comment](#)

Build solution :- Compiles code files (DLL and Exe) which are changed.

Rebuild :- Deletes all compiled filed and Compiles them again irrespective if the code has changed or not.

Clean solution :- Deletes all compiled files (DLL and Exe).

answered Jul 30 at 10:28



user2247744

15 6

[add a comment](#)

protected by Josh Crozier Aug 27 at 13:58

Thank you for your interest in this question. Because it has attracted low-quality answers, posting an answer now requires 10 [reputation](#) on this site.

Would you like to answer one of these [unanswered questions](#) instead?

Not the answer you're looking for? Browse other questions tagged [visual-studio](#) or [ask your own question](#).