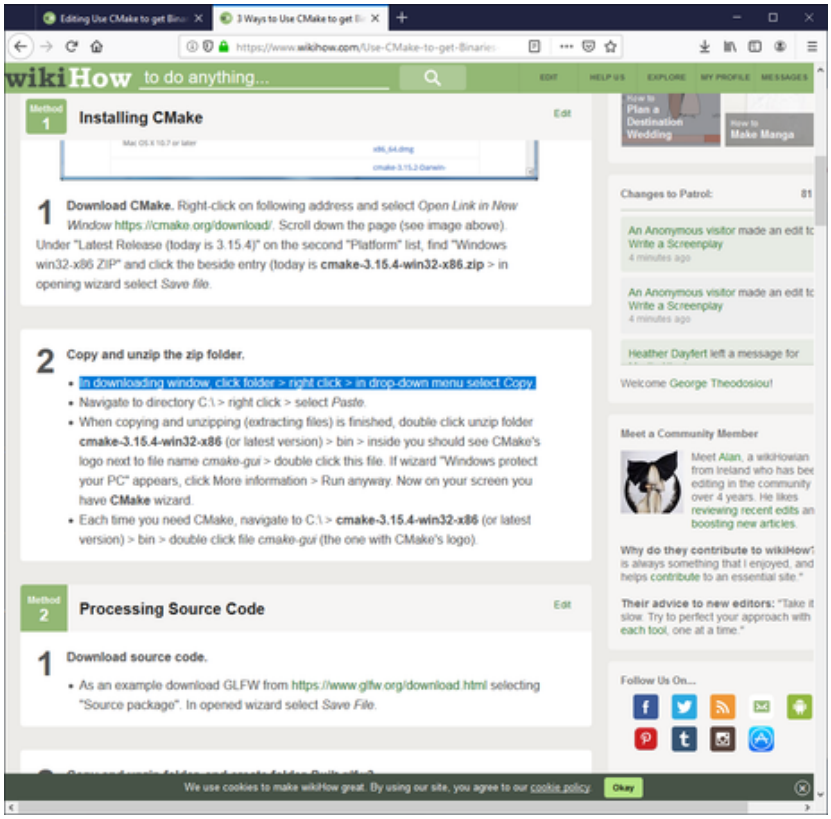
How to Use CMake to Get Binaries from Source Code

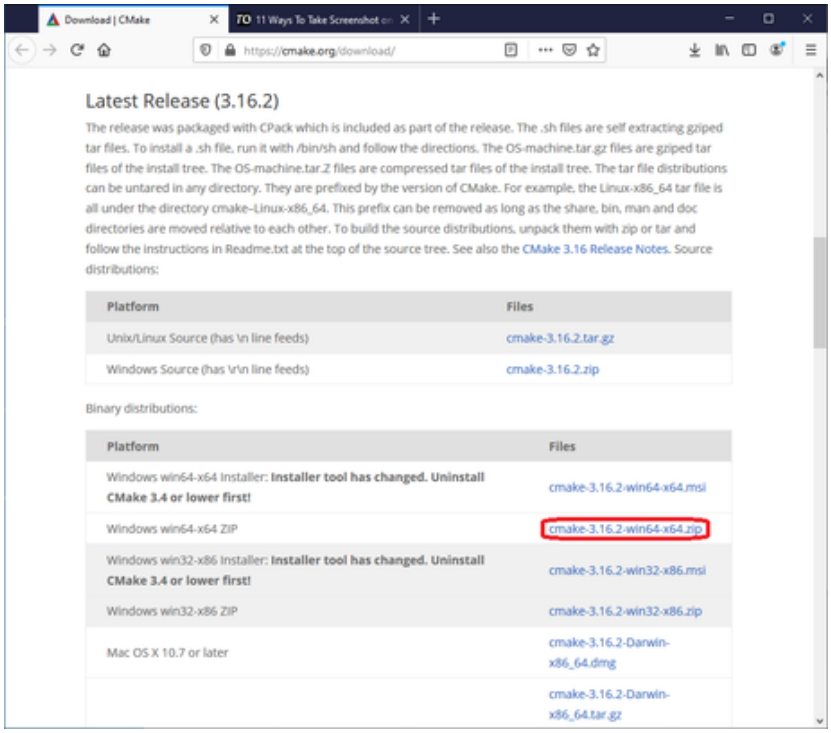
CMake is useful in compiling source code for get binaries. Compiling a library from the source code guarantees that the resulting library is perfectly tailored for your CPU/OS, a luxury pre-compiled binaries don't always provide. It is...

Method 1 of 3:

Installing CMake

1. 

Highlight what you expect to do. For example see picture above.

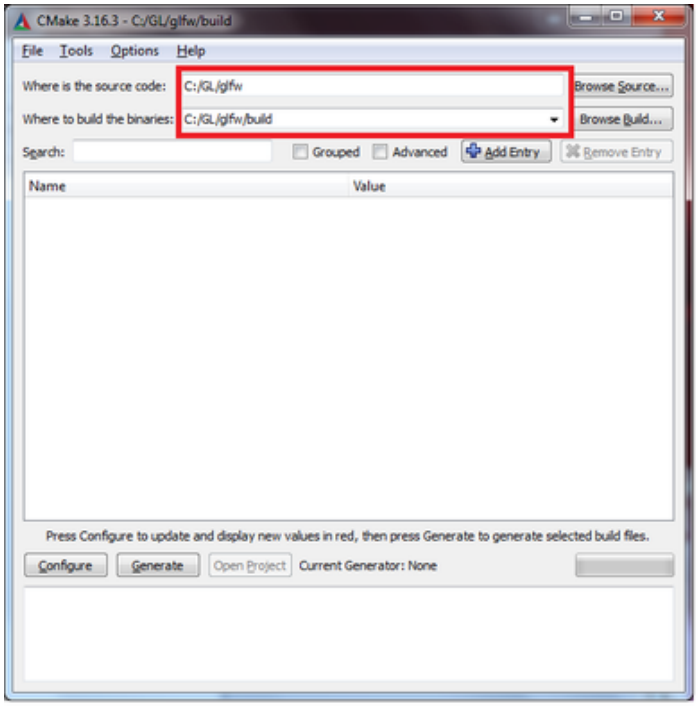
2. 

   Download CMake. Right-click on following address and select *Open Link in New Window* https://cmake.org/download/. Scroll down the page (see image above). Under "Latest Release (today is 3.16.2)" on the second "Platform" list, find "Windows win64-x64.ZIP" and click the beside entry (today is **cmake-3.16.2-win64-x64.zip** > in opening wizard select *Save file*.
3. **Copy and unzip the zip folder.**
   1. In downloading window, click folder > right click > in drop-down menu select *Copy*.
   2. Navigate to directory C: > right click > select *Paste*.
   3. When copying and unzipping (extracting files) is finished, double click unzip folder **cmake-3.16.2-win64-x64** (or latest version) > bin > inside you should see CMake's logo next to file name *cmake-gui* > double click this file. If wizard "Windows protect your PC" appears, click More information > Run anyway. Now on your screen you have **CMake** GUI.
   4. Each time you need CMake, navigate to C: > **cmake-3.16.2-win64-x64** (or latest version) > bin > double click file *cmake-gui* (the one with CMake's logo).

Method 2 of 3:

Processing Source Code

1. **In directory C, create folder *GL*.** In Windows *File Explorer* navigate to C: > inside, right click > select *New* > *Folder* > type: *GL* > hit ? Enter.
2. **Download source code.**
   1. As an example we use GLFW. Right-click on following address and select Open Link in New Window https://www.glfw.org/download.html. Select "Source package". In opened wizard select *Save File*.
3. **Copy and unzip folder.**
   1. In downloading window click zip folder glfw-3.3 (or latest version) > right click > select *Copy*.
   2. Navigate to C: > GL > right click > select *Paste*.

3. Now in directory C:GL, you have unzip folder *glfw-3.3*. Click twice on its name > delete name > type: *glfw* > hit ? Enter .
4. 

   Where is the source code. In CMake GUI first text field. Copy **C:/GL/glfw** and paste.
5. **Where to build the binaries.** In second text field. Copy **C:/GL/glfw/build** and paste.
6. **Configure and generate.** In CMake GUI, click Configure > in wizard *Create Directory* click Yes > select *Visual Studio 16 2019* > click Finish .
   1. When, in CMake GUI, you read: "Configuring done", click Generate . You should read: "Generating done".
   2. Close CMake GUI.
7. **Build your solution.**
   1. Navigate to C: > GL > glfw > build. Double click "GLFW.sln", or "GLFW", or "ALL_BUILD.vcxproj". An instance of Visual Studio appears. Wait until in main menu *Build* entry appears. Click it > "Build Solution".
   2. Wait till you read the last line in "Output" window: ========== Build: XX succeeded, 0 failed, 0 up-to-date, 2 skipped" ==========
      1. Number XX of "succeeded" changes in glfw versions. Today (31-12-2019) is 32.
   3. Navigate to C: > GL > glfw > build > src > Debug. Inside you should see file *glfw3.lib*.

Method 3 of 3:

Testing Binaries

1. **Create empty project.** In V.S. main menu click File > New > Project… > Create a new project > Empty project > Next .
   1. In *Project name* text box type: *Project-0.*
   2. Next to *Location* text box, click ... .

3. Navigate to C: > GL > Click *Select a folder*. The *Location* is C:GL.
4. Check box *Place solution and project in the same directory*.
5. Click Create.

2. **Add your source file to the Project.** In *Solution Explorer* wizard, right click the *Source Files* folder (the last one) > click Add > New Item…
    1. In the *Add New Item - Project-0* window, click *C++ File (.cpp)* (the first one) from the middle of the window. In the *Name* text box, type *Main.cpp*.
    2. The *Location* is C:GLProject-0.
    3. Click the Add button. The file will open in the main text editor but leave it blank for now.
3. **Configure project's *Properties*.** In *Solution Explorer* wizard, right click Project's name that is *Project-0* > select *Properties*.
    1. **(1) *Project-0 Property Pages* main menu.** In *Platform* entry select *x64* > click Configuration Manager....
        1. In *Active solution platform:* select *x64*.
        2. In *Platform* entry, *x64* is automatically selected.
        3. Click Close.
    2. **(2) *Additional Include Directories***. Click *C/C++* > *General* > In beside menu select the first one, *Additional Include Directories* > click the down arrow at the end of the field > click *Edit...* > first icon > three dots ...
        1. Navigate to C: > GL > glfw > include > click include > click Select a folder > click OK.
    3. **(3) *Additional Library Directories***. Double click *Linker* > click *General* > *Additional Library Directories* > click the down arrow at the end of the field > click *Edit...* > first icon > three dots ...
        1. Navigate to C: > GL > glfw > build > src > Debug > click Select a folder > click OK.
    4. **(4) *Additional Dependencies***. In *Linker* drop-down menu select *Input* > in beside menu select the first one, *Additional Dependencies* > click the down arrow at the end of the field > *Edit...* > copy **opengl32.lib; glfw3.lib** and paste in *Additional Dependencies* wizard's upper-most text box > click OK.
    5. **(5) Set System to SubSystem CONSOLE.** In *Linker* drop-down menu select *System* > in beside menu select the first one, *SubSystem* > click the down arrow at the end of the field > select *Console (/SUBSYSTEM:CONSOLE)*. Click Apply and OK.
4. **Test code.** Right-click on following address and select *Open Link in New Window* https://www.glfw.org/docs/3.0/quick.html > scroll down the page to last section: **Putting it together: A small GLFW application**, and copy code > go to Visual studio and paste it in *Main.cpp* code area. Hit Ctrl + F5. If everything gone well two windows appear: one black and other with a rotating colored triangle in black background. Good job.
5. **Correct any errors.** In "Error List" if you see error about
    1. file with extension .h go to Method 3, step 3, sub-step (2) and follow instructions.
    2. file with extension .lib go Method 3, step 3, sub-step (3), and follow instructions. Also to sub-step (4).
    3. "entry point must be defined" go to Method 3, step 3, sub-step (5), **Set "SubSystem" to "CONSOLE"**, and follow instructions.
    4. For other errors, if you can't correct them, close Visual Studio > delete project folder *Project-0* which lives in C:GL > open Visual Studio > repeat set up from part *Method 2*.