

Installing CLion and Clang

Table of contents

Install/Confirm Clang compiler.....	3
Install CLion IDE	5
Test installation.....	7
Configure linter.....	9

Install/Confirm Clang compiler



For MacOS

- Open a new terminal window.
- Enter the command `clang --version` to confirm that you already have Clang installed on your machine. You should get a similar message followed if Clang is installed:

```
Apple LLVM version 9.0.0 (clang-900.0.39.2)
Target: x86_64-apple-darwin17.3.0
Thread model: posix
```

If you want to install or update Clang, enter the command `command xcode-select --install`.



For Windows

During the installation of MSYS2, you must have Internet access as the installer will be downloading files during the installation process.

- Open the MSYS2 website at: msys2.org
- On the homepage, download the 64-bit installer (the .exe file that starts with "msys2-x86_64") and run it.
- On the **Setup – MSYS2 64bit** dialog window, click **Next**.
- On the **Installation Folder** dialog window, keep the default folder (should be `c:\msys64`) and click **Next**.
- On the **Start Menu shortcuts** dialog window, keep the default name (should be `MSYS 64bit`) and click **Next**.
- On the **Installing** dialog window, the installation may take a few moments. When complete, a message should be shown stating that installation has finished. Click **Next**.
- On the **Finished** dialog window, uncheck **Run MSYS2 64bit now** and click **Finish**.

Make sure there are no MSYS2 command prompt windows that are still open.

- Find the MSYS2 menu in the Start menu and select **MSYS2 MSYS** to open an MSYS command prompt window.
- On the MSYS command prompt, type the following command and press **Enter**.

```
pacman --noconfirm -Syu && pacman --noconfirm -S mingw-w64-x86_64-toolchain mingw-w64-x86_64-clang
```

Note: you may copy the command above. To paste it into the command prompt, right click in the command prompt window and select **Paste**.

```
vanderbilt MSYS ~
$ pacman --noconfirm -Syu && pacman --noconfirm -S mingw-w64-x86_64-toolchain mingw-w64-x86_64-clang
```

- Once the installation of the MinGW toolchain and Clang is complete, you should see the following text in the MSYS command prompt. Type `exit` and press **Enter**. Note that the installation may take several minutes.

```
vanderbilt MSYS ~
(28/48) installing mingw-w64-x86_64-libtasn1 [#####] 100%
(29/48) installing mingw-w64-x86_64-p11-kit [#####] 100%
(30/48) installing mingw-w64-x86_64-ca-certificates [#####] 100%
(31/48) installing mingw-w64-x86_64-openssl [#####] 100%
(32/48) installing mingw-w64-x86_64-termcap [#####] 100%
(33/48) installing mingw-w64-x86_64-readline [#####] 100%
(34/48) installing mingw-w64-x86_64-tcl [#####] 100%
(35/48) installing mingw-w64-x86_64-sqlite3 [#####] 100%
(36/48) installing mingw-w64-x86_64-tk [#####] 100%
(37/48) installing mingw-w64-x86_64-xz [#####] 100%
(38/48) installing mingw-w64-x86_64-python [#####] 100%
(39/48) installing mingw-w64-x86_64-xxhash [#####] 100%
(40/48) installing mingw-w64-x86_64-gdb [#####] 100%
(41/48) installing mingw-w64-x86_64-libmangle-git [#####] 100%
(42/48) installing mingw-w64-x86_64-make [#####] 100%
(43/48) installing mingw-w64-x86_64-pkg-config [#####] 100%
(44/48) installing mingw-w64-x86_64-tools-git [#####] 100%
(45/48) installing mingw-w64-x86_64-winstorecompat-git [#####] 100%
(46/48) installing mingw-w64-x86_64-z3 [#####] 100%
(47/48) installing mingw-w64-x86_64-llvm [#####] 100%
(48/48) installing mingw-w64-x86_64-clang [#####] 100%

vanderbilt MSYS ~
$
```

Install CLion IDE

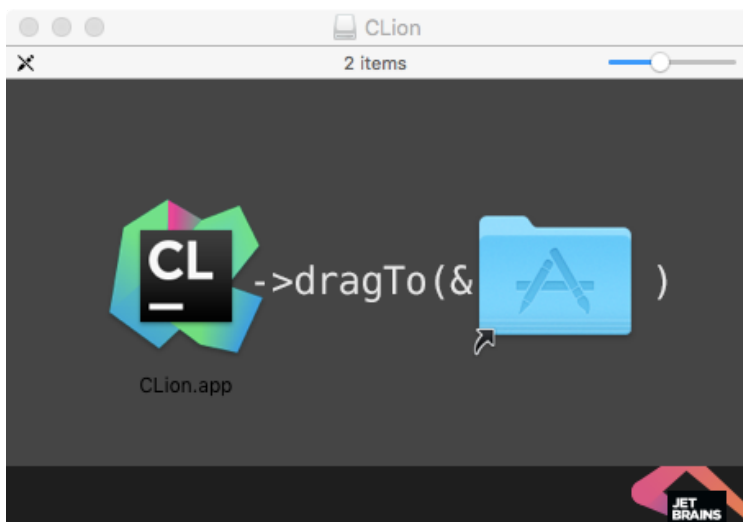
We will be using the CLion IDE developed by JetBrains. If you haven't already done so, you will need to obtain an individual license for JetBrains products. For students, this license is free.

- Visit the CLion website at: jetbrains.com/clion.
- On the CLion homepage, click **Buy** near top of the page.
- On the *Subscription options & Pricing* page, click on the **Special Offers** tab.
- Under the *For students and teachers* header, click **Learn more**.
- On the *Free Educational Licenses* page, make sure the **For students and teachers** option is selected. Scroll down and click on the **Apply Now** button.
- Apply for a license with the **University email address** option. You must use your Vanderbilt email address to receive the free license. You will receive a confirmation email. Click on the **Confirm Request** link in the email you received. Follow any instructions on the page you visit. As you go through this process, be sure to make notes of your user name, password, license ID, and your activation code.
- You will receive another email confirming your student license. Click on the provided link in the email to activate your educational license and save your activation code. On this page, there are links to download various products. Click on the **CLion** link that will take you back to the CLion home page.



For MacOS

- Click on the **Download** button near the top-right of the page on the JetBrains website.
- On the **Download** page, download the **macOS** version of CLion and open it. Make sure to pick the correct DMG file based on the chip on your computer.
- Drag the CLion icon into your **Applications** folder as shown.



- You can now unmount the CLion DMG and throw it in the trash.
- Now, go to your **Applications** folder and open **CLion**. You may get the dialog window stating that "**CLion**" is an app downloaded from the Internet. If you do, click **Open**.
- When starting CLion for the first time, you will be asked whether to accept the JetBrains User Agreement. If you do not want to continue installation, click **Exit**. Otherwise, check the checkbox and click **Continue**.
- You will then be presented with the following **Licenses** dialog window. Check the **Activation code** button and copy-n-paste your activation code that you saved into the text box, and click **OK**. You also to select the **JB Account** option (pictured below) and enter your JetBrains username/password.



For Windows

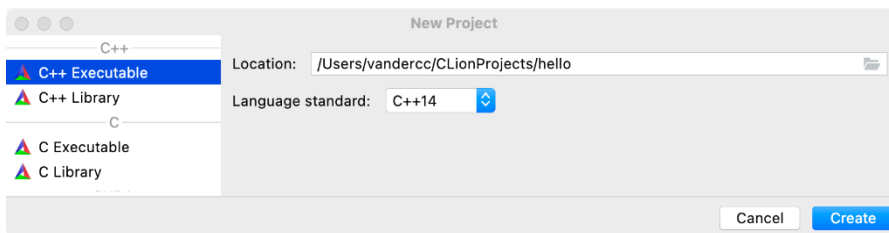
- Now click on the **Download** button near the top-right of the page on the JetBrains website.
- On the **Download** page, download the Windows version of CLion and save the installer to your computer.
- Run the CLion installer you just saved. Click **Next** on the **Welcome to CLion Setup** dialog window.
- On the **Choose Install Location** dialog window, accept the default folder by clicking **Next**.
- On the **Installation Options** dialog window, select whether you want to create a desktop shortcut and associate related file extensions with CLion. When done, click **Next**.
- On the **Choose Start Menu Folder** dialog window, the name of the new menu folder for CLion will be listed. Click **Install**.
- When the install has completed, check the **Run CLion** box and click **Finish**.
- When starting CLion for the first time, you will be asked whether to accept the JetBrains User Agreement. If you do not want to continue installation, click **Exit**. Otherwise, check the checkbox and click **Continue**.
- You will then be presented with the following **License Activation** dialog window. Check the **Activation code** button and copy-n-paste your activation code that you saved into the text box, and click **OK**. You also to select the **JB Account** option (pictured below) and enter your JetBrains username/password.

Test installation



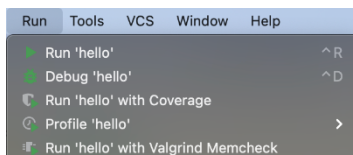
For MacOS

- After the previous step, the **Welcome to CLion** dialog window should be displayed. Click **New Project**.
- Enter a folder name (something other than "test") and a location for the project (use a directory you can later find or accept the default), and press **Create**. In the example below, the project name is **Hello** with the associated located at **C:\Users\vandercc\CLionProjects\hello**.



The IDE will now open up with two tabs: CMakeLists.txt and main.cpp. The main.cpp tab will be displayed and it will contain a simple Hello World program.

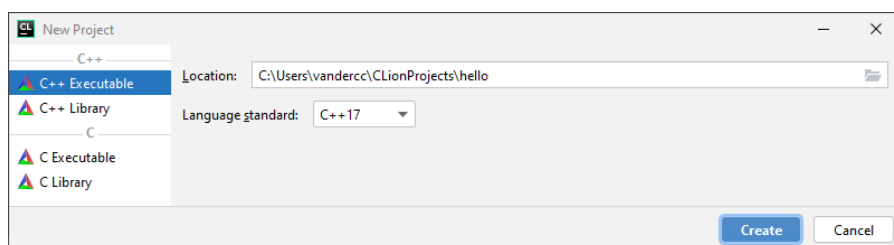
- In the **Open Project Wizard** dialog window, select the defaults by clicking **OK**.
- From the menu options, select **Run | Run 'hello'**. The text "Hello, World!" at the bottom.





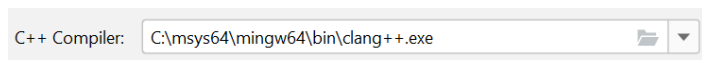
For Windows

- After the previous step, the **Welcome to CLion** dialog window should be displayed. Click **New Project**.
- Enter a folder name (something other than "test") and a location for the project (use a directory you can later find or accept the default), and press **Create**. In the example below, the project name is **hello** with the associated located at **C:\Users\vandercc\CLionProjects\hello**.

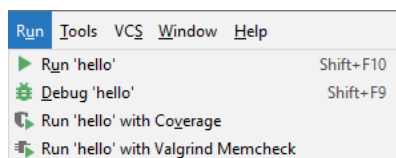


The IDE will now open up with two tabs: CMakeLists.txt and main.cpp. The main.cpp tab will be displayed and it will contain a simple Hello World program.

- In the **Open Project Wizard** dialog window, **MinGW** should be selected. The C++ compiler will be set to GCC. To use the Clang compiler, click on the folder icon to the right of **C++ Compiler**. Search for and select the **C:\msys64\mingw64\bin\clang++.exe** file. The listing of that file should now be displayed on the dialog window. Click **OK**.



- From the menu options, select **Run | Run 'hello'**. The text "Hello, World!" at the bottom.



Configure linter

For the C++ assignments, one of the three builds that is processed on GitHub is *linter*. Linter checks that the code that you committed on GitHub conforms to the *WebKit* code standard (more information at: webkit.org/code-style-guidelines). You can use **clang-format**, which can automatically update your code's formatting to comply with the code style guidelines. Hence, it is highly recommended to run **clang-format** before you commit to GitHub.

After completing the following steps, you can call **clang-format** from inside CLion by going to **Tools | External Tools** and select **clang-format** (the name used in the instructions below). The program will automatically update your code (if needed) to conform to the specified code standard, which in this case is WebKit. Note that the updates are only related to code style formatting.



For MacOS

Note: you may have to `brew install clang-format` for the linter to work.

- In CLion, go to **CLion | Preferences | Tools | External Tools** and click the + in the top left.
- For the **Name** field, type `clang-format`
- For the **Program** field, type `bash`
- The **Arguments** field should be:

```
-c "cd '$ProjectFileDir$' && /usr/bin/find -E ./ -regex '\.\/\/(src|include|tests).*\.(h|cpp)' -exec clang-format -i -style=file {} \;"
```

- For the **Working Directory** field, type `$ProjectFileDir$` (you literally type `$ProjectFileDir$` in the textbox including the dollar signs)



For Windows

- In CLion, go to **Settings | Tools | External Tools** and click the + in the top left.
- For the **Name** field, type `clang-format`
- For the **Program** field, type `C:\msys64\usr\bin\sh.exe`
- The **Arguments** field should be:

```
-c "cd '$ProjectFileDir$' && /usr/bin/find . -regex '.*\.(h|cpp\)' -exec /c/msys64/mingw64/bin/clang-format -i -style=file {} \; ; /usr/bin/find . -regex '.*\.(h|cpp\).*\..TMP' -exec /usr/bin/rm {} \;"
```

- The **Working Directory** field, type `$ProjectFileDir$` (you literally type `$ProjectFileDir$` in the textbox including the dollar signs)



For Linux

- In CLion, go to **Settings | Tools | External Tools** and click the + in the top left.
- For the **Name** field, type `clang-format`
- For the **Program** field, type `bash`
- The **Arguments** field should be:

```
-c "cd '$ProjectFileDir$' && /usr/bin/find -E ./ -regex '\.\/\/(src|include|tests).*\.(h|cpp)' -exec clang-format -i -style=file {} \;"
```

The **Working Directory** field, type `$ProjectFileDir$` (you literally type `$ProjectFileDir$` in the textbox including the dollar signs)