

Static build for Windows MSVC

Creating a static build for Windows (native binaries with MSVC)

- Main documentation
 - [Windows build overview](#)
 - [configure.bat options](#)
- Obtain the sources either from the official offline installer download site (.zip archive), or github:
 - [Offline installers](#)
 - [Github](#)
- Ensure that the following tools are on your path:

Build Tools

Tool	Supported Versions	Description
CMake	Version 3.16 and newer (3.17 and newer for - debug-and-release builds 3.21 and newer for - static builds).	Required for configuring the Qt build. Available in Qt Online Installer and on cmake.org .
Ninja	-	Recommended tool for building Qt. Available in Qt Online Installer and on ninja-build.org .
Python	Version 3	Required build tool. Windows installers are available on python.org , or from the Microsoft Store .

The executables `cmake.exe`, `ninja.exe`, and `python.exe` must be available in your build environment. Add the respective directory to your `PATH` environment variable.

- Open a terminal, either
 - Use a Visual Studio developer powershell or `cmd.exe` terminal (available from the application menu or in Windows Terminal)
 - A standard powershell or `cmd.exe` terminal with the `vcvarsall.bat` command:

```
"C:\Program Files\Microsoft Visual  
Studio\2022\Professional\VC\Auxiliary\Build\vcvarsall.bat" amd64
```

- Execute the following command from the shadow directory (be sure to set the `-prefix` if you want it to install to a different location):

```
> <SOURCE_DIRECTORY>/configure.bat -debug-and-release -static -opensource -  
confirm-license -prefix C:\Qt\6.6.3\Build-Static-MSVC -platform win32-msvc -  
nomake examples -nomake tests  
> ninja  
> ninja install
```

- Create the new build kit in Qt Creator

Full configure_args.txt

CONFIGURE BUILD ENVIRONMENT

Developer powershell or

"C:\Program Files\Microsoft Visual

Studio\2022\Professional\VC\Auxiliary\Build\vcvarsall.bat" amd64

CONFIGURE QT

MSVC

./configure.bat -debug -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MSVC-Debug -platform win32-msvc -nomake examples -nomake tests

./configure.bat -release -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MSVC-Release -platform win32-msvc -nomake examples -nomake tests

./configure.bat -debug-and-release -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MSVC -platform win32-msvc -nomake examples -nomake tests

MINGW

./configure.bat -debug -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MINGW-Debug -platform win32-g++ -nomake examples -nomake tests

./configure.bat -release -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MINGW-Release -platform win32-g++ -nomake examples -nomake tests

./configure.bat -debug-and-release -static -opensource -confirm-license -prefix

C:\Qt\6.6.3\Build-Static-MINGW -platform win32-g++ -nomake examples -nomake tests

BUILD AND INSTALL

ninja

ninja install

#####

DIRECTORIES

Source code directory: e.g. C:\Qt\SourceDistribution\qt-everywhere-src-6.6.3-ORIGINAL

Build (shadow) directory (artefacts such as .obj): e.g.

C:\Qt\SourceDistribution\SHADOW-MSVC

Installation directory: e.g. C:\Qt\6.6.3\Build-Static-MSVC

CONFIGURE BUILD ENVIRONMENT

Developer powershell or cmd.exe with "C:\Program Files\Microsoft Visual

Studio\2022\Professional\VC\Auxiliary\Build\vcvarsall.bat" amd64

SHADOW BUILD AND INSTALL

cd to shadow_directory

source_directory\configure.bat -debug-and-release -static -opensource -confirm-license -prefix C:\Qt\6.6.3\Build-Static-MSVC -platform win32-msvc -nomake examples -nomake tests

ninja

ninja install

CLEAN

"C:/Qt/Tools/mingw1310_64/bin/mingw32-make.exe" clean

```
./configure.bat -redo  
May need to also delete CMakeCache.txt
```