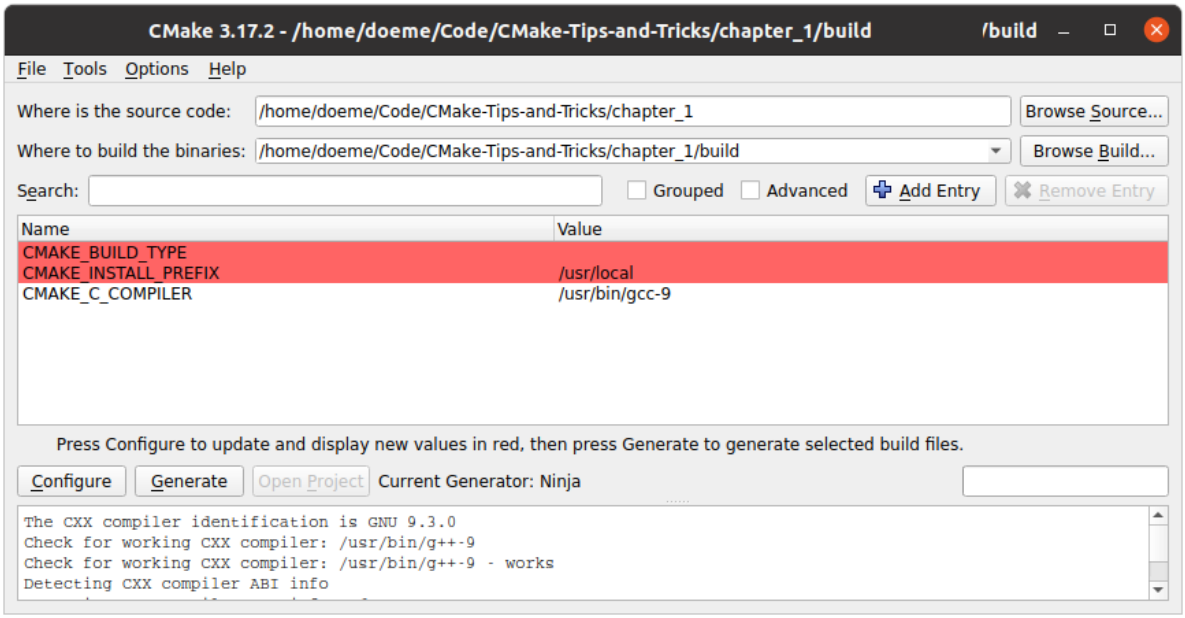
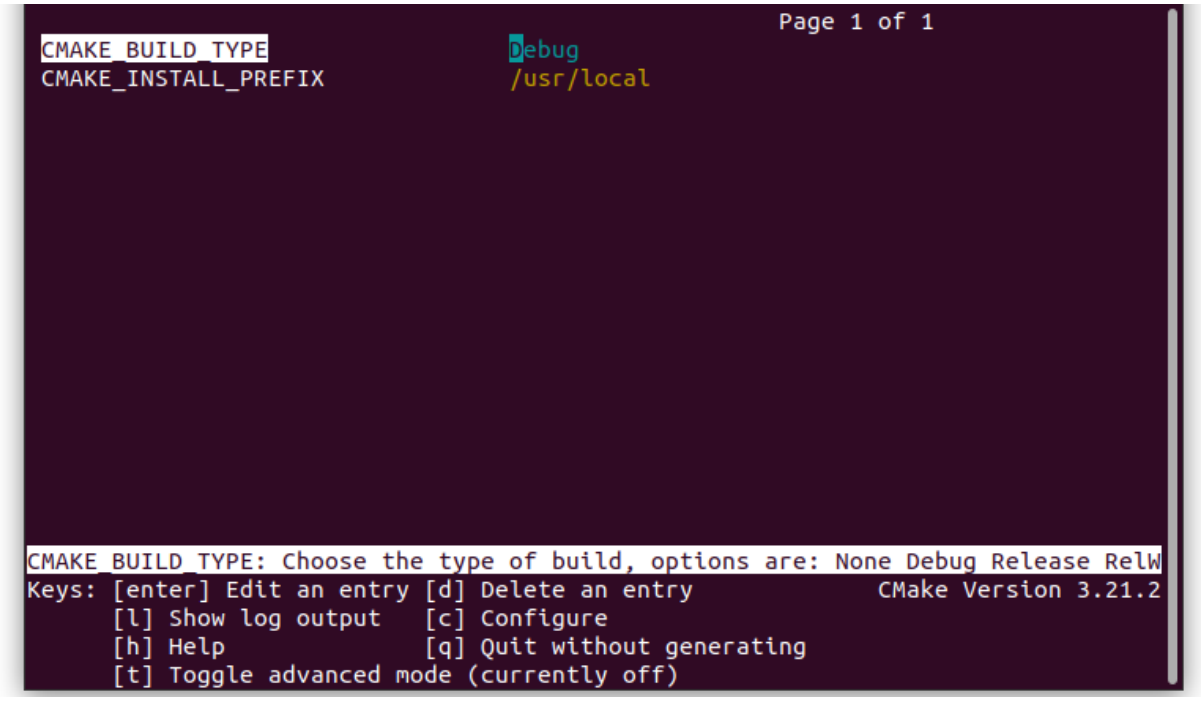
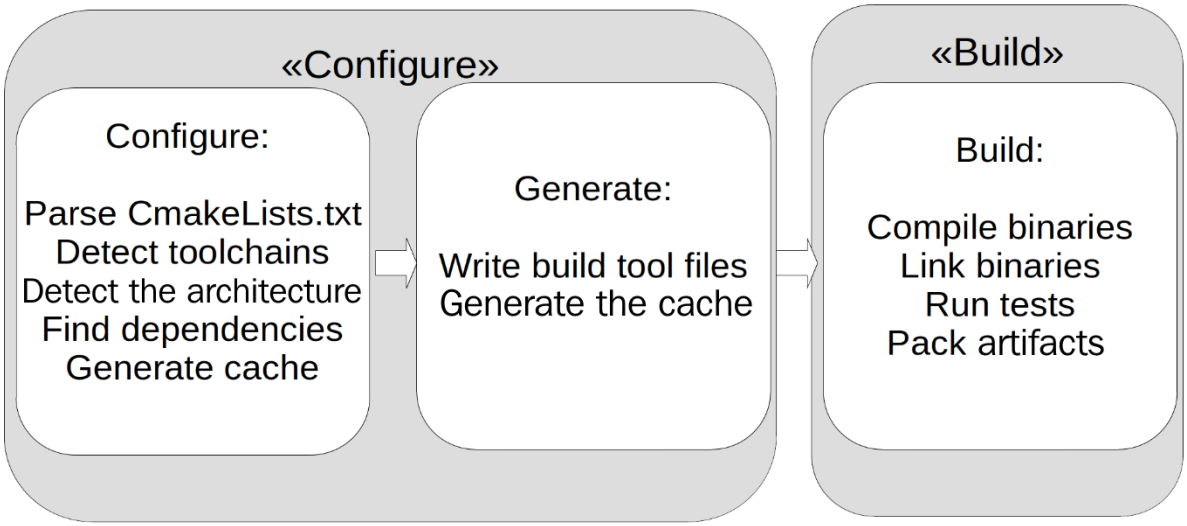
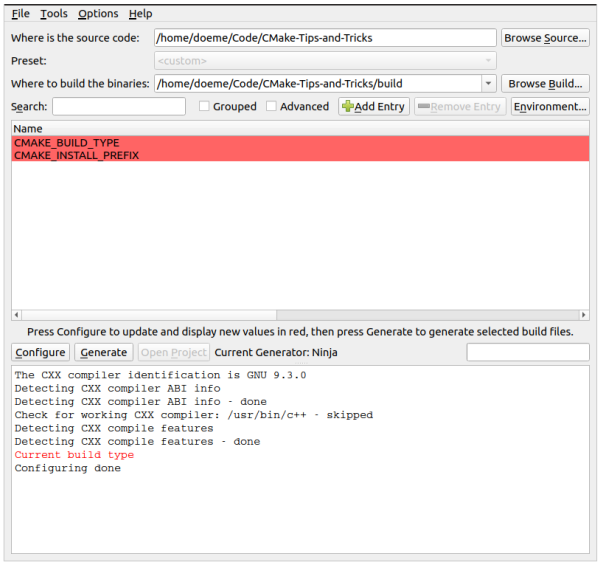
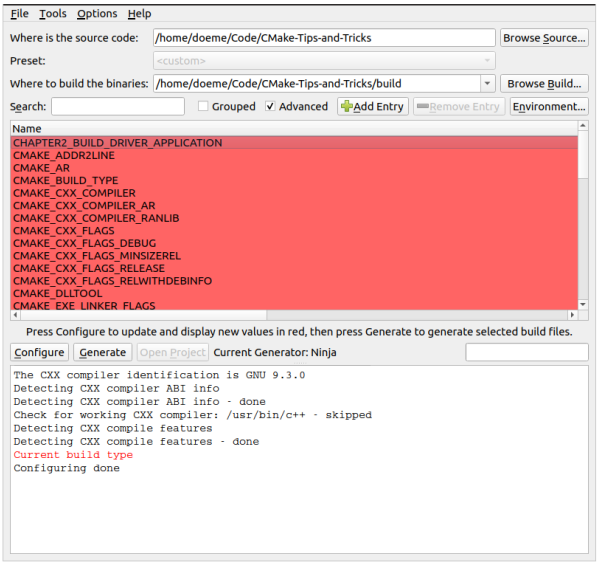
Chapter 1: Kickstarting CMake



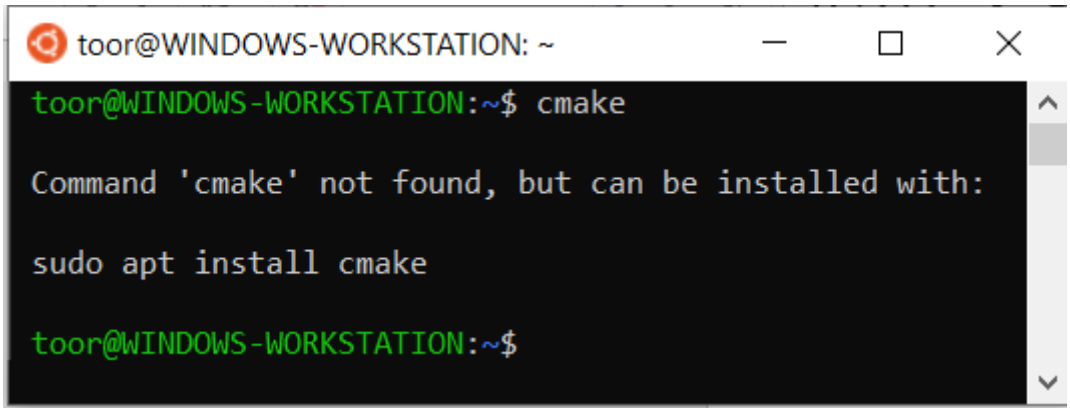


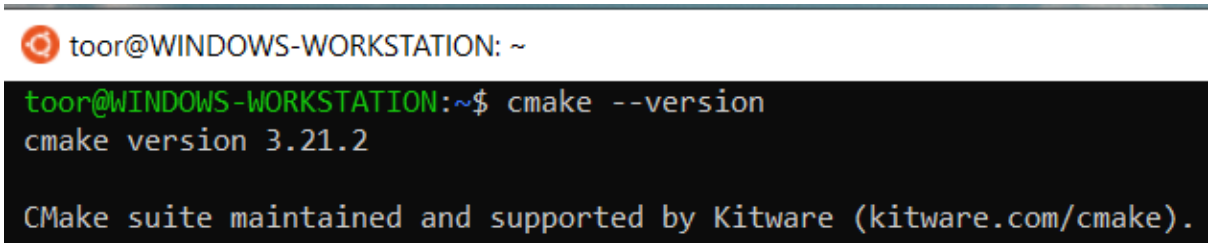

«Configure»
Configure:
Parse CmakeLists.txt
Detect toolchains
Detect the architecture
Find dependencies
Generate cache
Generate:
Write build tool files
Generate the cache
«Build»
Build:
Compile binaries
Link binaries
Run tests
Pack artifacts


File Tools Options Help
Where is the source code: /home/doeme/Code/CMake-Tips-and-Tricks Browse Source...
Preset: <custom>
Where to build the binaries: /home/doeme/Code/CMake-Tips-and-Tricks/build Browse Build...
Search: Grouped Advanced Add Entry Remove Entry Environment...
Name
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX
Press Configure to update and display new values in red, then press Generate to generate selected build files.
Configure Generate Open Project Current Generator: Ninja
The CXX compiler identification is GNU 9.3.0
Detecting CXX compiler ABI info
Detecting CXX compiler ABI info - done
Check for working CXX compiler: /usr/bin/c++ - skipped
Detecting CXX compile features
Detecting CXX compile features - done
Current build type
Configuring done

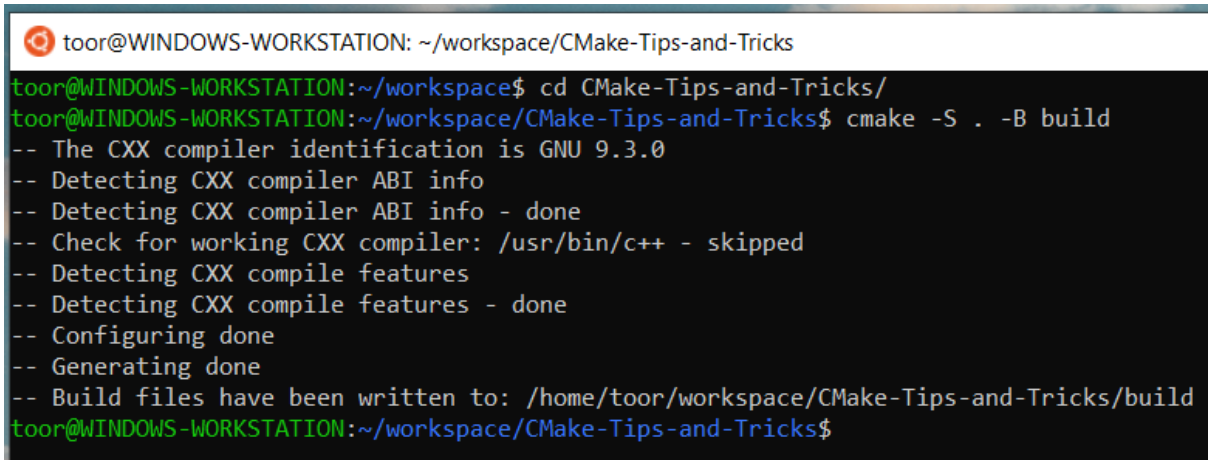

File Tools Options Help
Where is the source code: /home/doeme/Code/CMake-Tips-and-Tricks Browse Source...
Preset: <custom>
Where to build the binaries: /home/doeme/Code/CMake-Tips-and-Tricks/build Browse Build...
Search: Grouped Advanced Add Entry Remove Entry Environment...
Name
CHAPTER2_BUILD_DRIVER_APPLICATION
CMAKE_ADDR2LINE
CMAKE_AR
CMAKE_BUILD_TYPE
CMAKE_CXX_COMPILER
CMAKE_CXX_COMPILER_AR
CMAKE_CXX_COMPILER_RANLIB
CMAKE_CXX_FLAGS
CMAKE_CXX_FLAGS_DEBUG
CMAKE_CXX_FLAGS_MINSIZEREL
CMAKE_CXX_FLAGS_RELEASE
CMAKE_CXX_FLAGS_RELWITHDEBINFO
CMAKE_DLLTOOL
CMAKE_EXE_LINKER_FLAGS
Press Configure to update and display new values in red, then press Generate to generate selected build files.
Configure Generate Open Project Current Generator: Ninja
The CXX compiler identification is GNU 9.3.0
Detecting CXX compiler ABI info
Detecting CXX compiler ABI info - done
Check for working CXX compiler: /usr/bin/c++ - skipped
Detecting CXX compile features
Detecting CXX compile features - done
Current build type
Configuring done

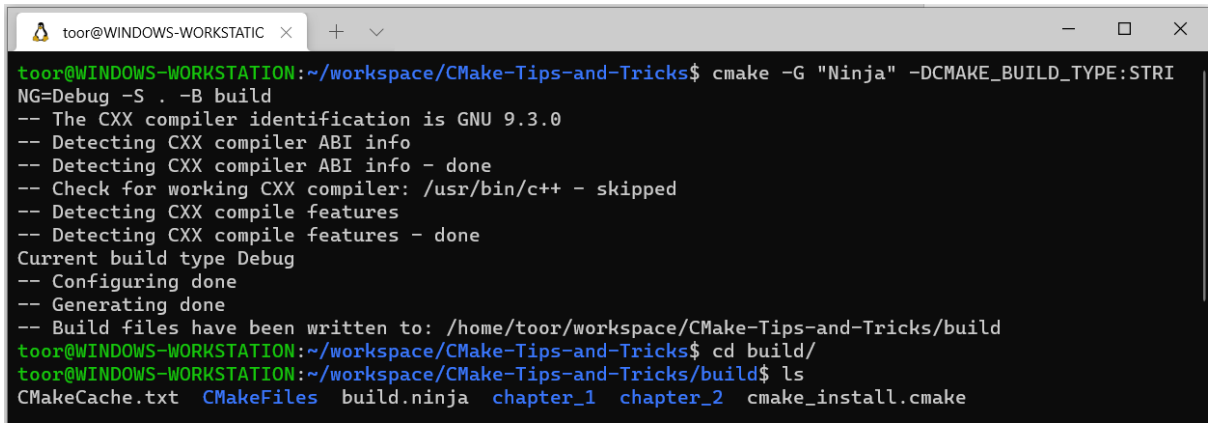
Chapter 2: Accessing CMake in Best Ways

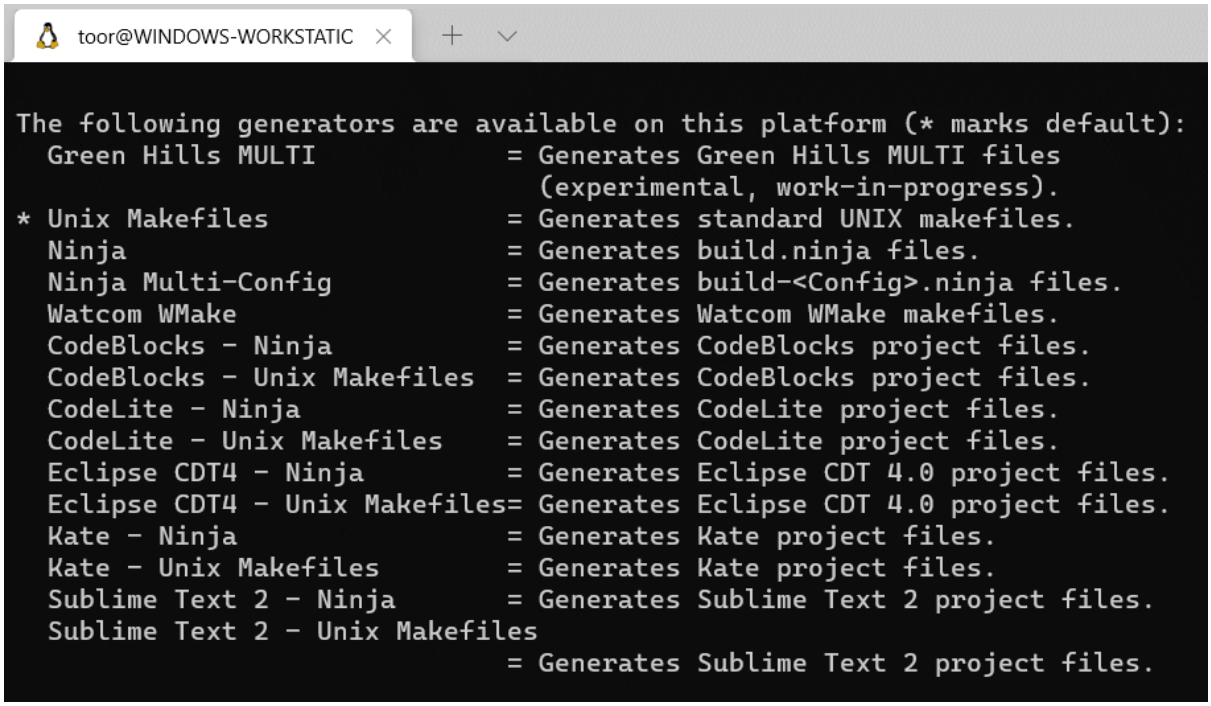


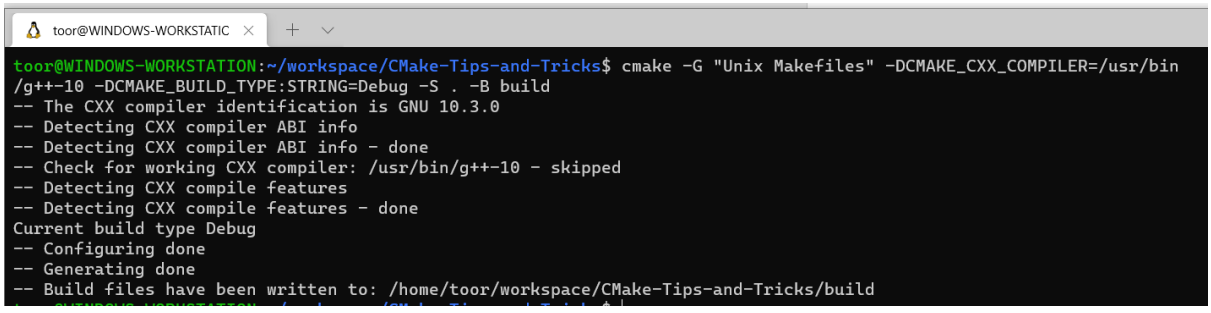


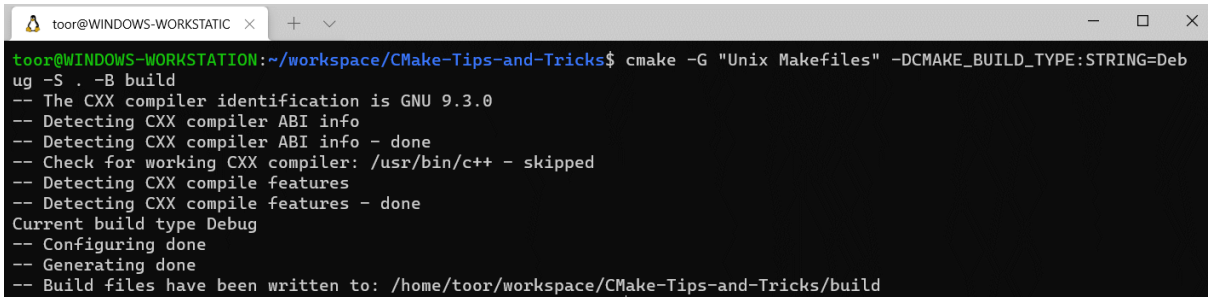

```
toor@WINDOWS-WORKSTATION:~/workspace$ git clone https://github.com/PacktPublishing/CMake-Tips-and-Tricks.git
Cloning into 'CMake-Tips-and-Tricks'...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (59/59), done.
remote: Total 95 (delta 32), reused 85 (delta 26), pack-reused 0
Unpacking objects: 100% (95/95), 10.49 KiB | 38.00 KiB/s, done.
toor@WINDOWS-WORKSTATION:~/workspace$ ls
CMake-Tips-and-Tricks
```

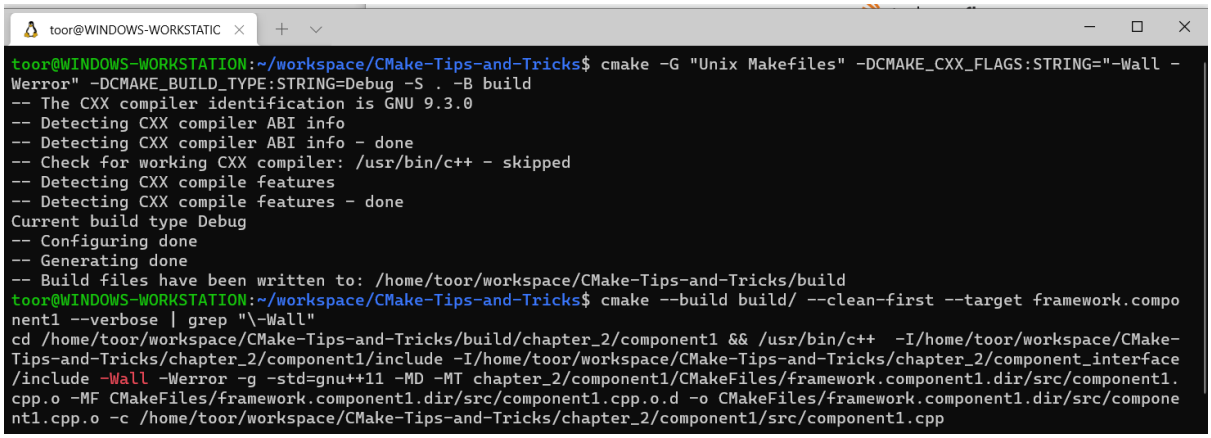


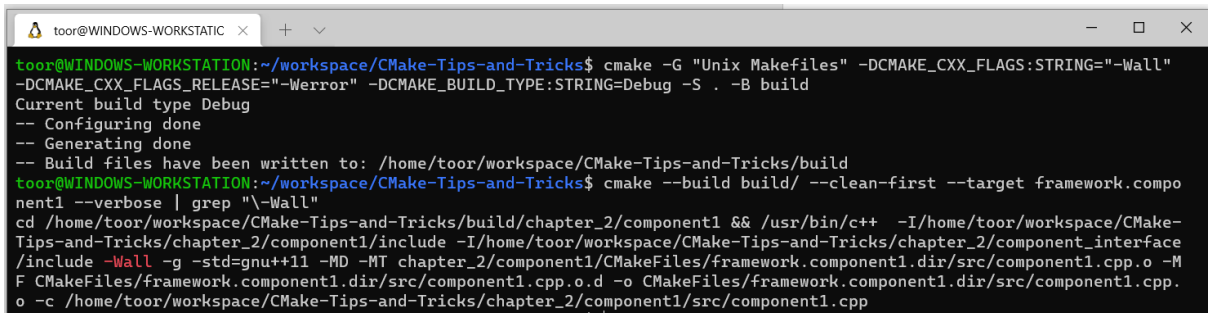

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Ninja" -DCMAKE_BUILD_TYPE:STRI
NG=Debug -S . -B build
-- The CXX compiler identification is GNU 9.3.0
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
Current build type Debug
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cd build/
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks/build$ ls
CMakeCache.txt CMakeFiles build.ninja chapter_1 chapter_2 cmake_install.cmake

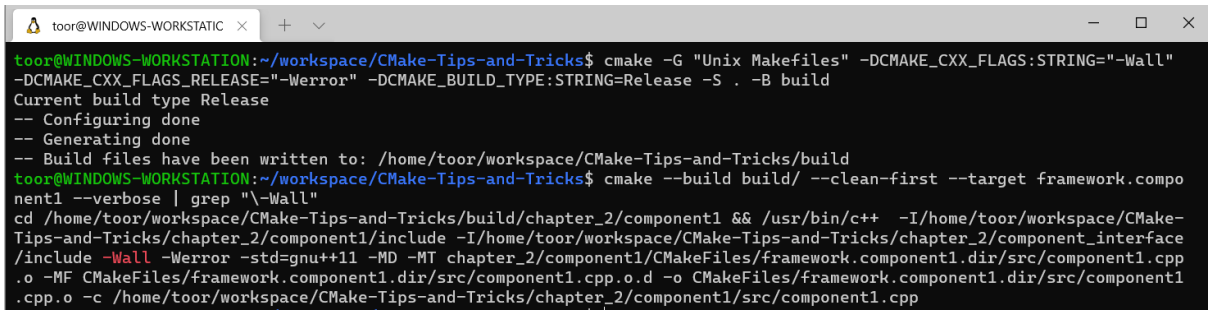

toor@WINDOWS-WORKSTATIC
The following generators are available on this platform (* marks default):
Green Hills MULTI = Generates Green Hills MULTI files
(experimental, work-in-progress).
* Unix Makefiles = Generates standard UNIX makefiles.
Ninja = Generates build.ninja files.
Ninja Multi-Config = Generates build-<Config>.ninja files.
Watcom WMake = Generates Watcom WMake makefiles.
CodeBlocks - Ninja = Generates CodeBlocks project files.
CodeBlocks - Unix Makefiles = Generates CodeBlocks project files.
CodeLite - Ninja = Generates CodeLite project files.
CodeLite - Unix Makefiles = Generates CodeLite project files.
Eclipse CDT4 - Ninja = Generates Eclipse CDT 4.0 project files.
Eclipse CDT4 - Unix Makefiles= Generates Eclipse CDT 4.0 project files.
Kate - Ninja = Generates Kate project files.
Kate - Unix Makefiles = Generates Kate project files.
Sublime Text 2 - Ninja = Generates Sublime Text 2 project files.
Sublime Text 2 - Unix Makefiles
= Generates Sublime Text 2 project files.

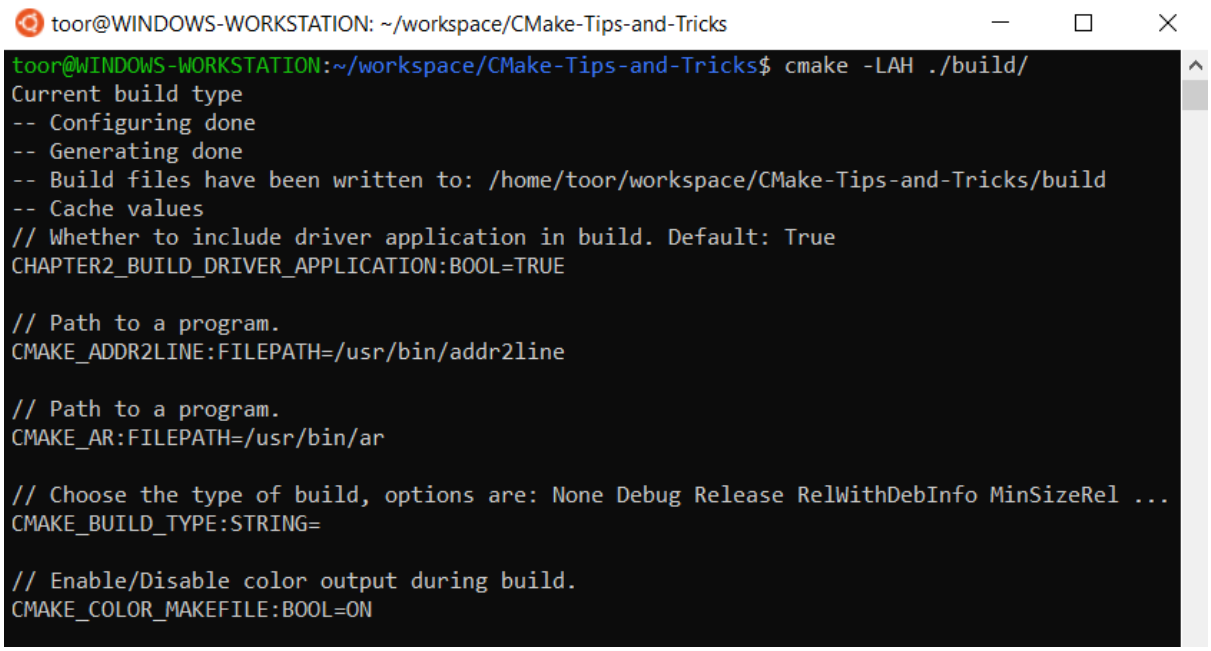

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Unix Makefiles" -DCMAKE_CXX_COMPILER=/usr/bin
/g++-10 -DCMAKE_BUILD_TYPE:STRING=Debug -S . -B build
-- The CXX compiler identification is GNU 10.3.0
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/g++-10 - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
Current build type Debug
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build

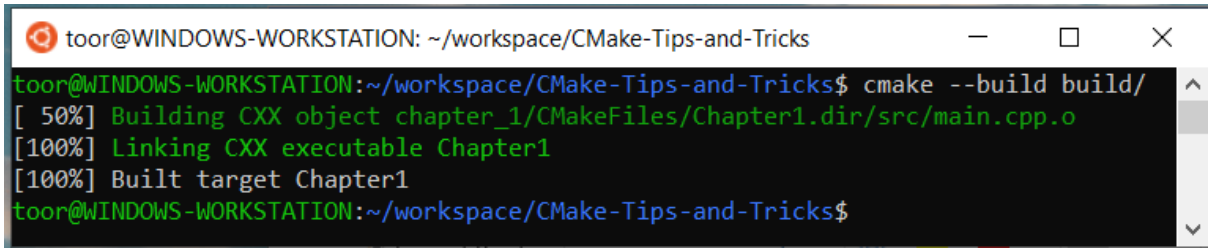

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Unix Makefiles" -DCMAKE_BUILD_TYPE:STRING=Debug -S . -B build
-- The CXX compiler identification is GNU 9.3.0
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
Current build type Debug
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build

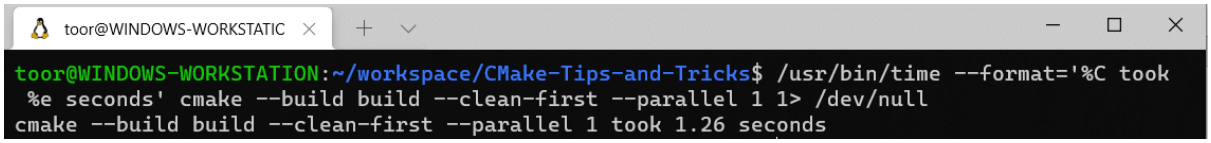

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Unix Makefiles" -DCMAKE_CXX_FLAGS:STRING="-Wall -Werror" -DCMAKE_BUILD_TYPE:STRING=Debug -S . -B build
-- The CXX compiler identification is GNU 9.3.0
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
Current build type Debug
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake --build build/ --clean-first --target framework.component1 --verbose | grep "\-Wall"
cd /home/toor/workspace/CMake-Tips-and-Tricks/build/chapter_2/component1 && /usr/bin/c++ -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/include -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component_interface/include -Wall -Werror -g -std=gnu++11 -MD -MT chapter_2/component1/CMakeFiles/framework.component1.dir/src/component1.cpp.o -MF CMakeFiles/framework.component1.dir/src/component1.cpp.o.d -o CMakeFiles/framework.component1.dir/src/component1.cpp.o -c /home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/src/component1.cpp

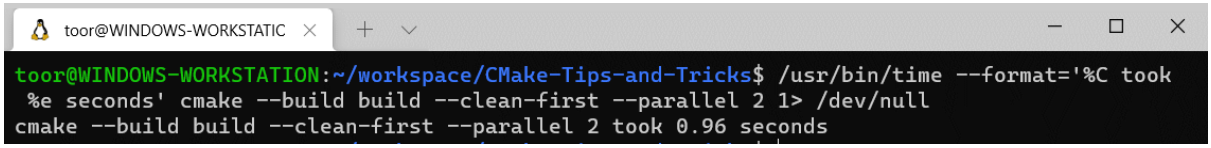

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Unix Makefiles" -DCMAKE_CXX_FLAGS:STRING="-Wall" -DCMAKE_CXX_FLAGS_RELEASE="-Werror" -DCMAKE_BUILD_TYPE:STRING=Debug -S . -B build
Current build type Debug
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake --build build/ --clean-first --target framework.component1 --verbose | grep "\-Wall"
cd /home/toor/workspace/CMake-Tips-and-Tricks/build/chapter_2/component1 && /usr/bin/c++ -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/include -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component_interface/include -Wall -g -std=gnu++11 -MD -MT chapter_2/component1/CMakeFiles/framework.component1.dir/src/component1.cpp.o -MF CMakeFiles/framework.component1.dir/src/component1.cpp.o.d -o CMakeFiles/framework.component1.dir/src/component1.cpp.o -c /home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/src/component1.cpp

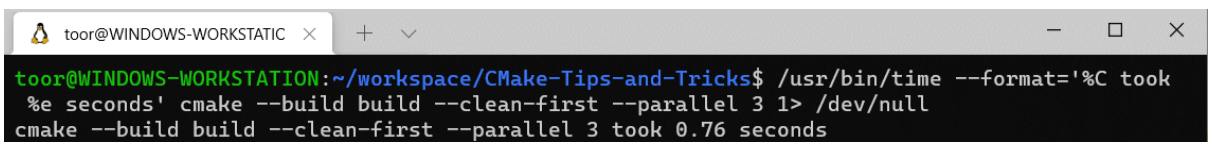

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -G "Unix Makefiles" -DCMAKE_CXX_FLAGS:STRING="-Wall" -DCMAKE_CXX_FLAGS_RELEASE="-Werror" -DCMAKE_BUILD_TYPE:STRING=Release -S . -B build
Current build type Release
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake --build build/ --clean-first --target framework.component1 --verbose | grep "\-Wall"
cd /home/toor/workspace/CMake-Tips-and-Tricks/build/chapter_2/component1 && /usr/bin/c++ -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/include -I/home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component_interface/include -Wall -Werror -std=gnu++11 -MD -MT chapter_2/component1/CMakeFiles/framework.component1.dir/src/component1.cpp.o -MF CMakeFiles/framework.component1.dir/src/component1.cpp.o.d -o CMakeFiles/framework.component1.dir/src/component1.cpp.o -c /home/toor/workspace/CMake-Tips-and-Tricks/chapter_2/component1/src/component1.cpp

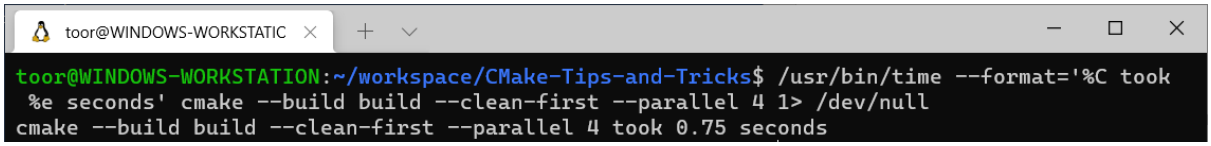

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake -LAH ./build/
Current build type
-- Configuring done
-- Generating done
-- Build files have been written to: /home/toor/workspace/CMake-Tips-and-Tricks/build
-- Cache values
// Whether to include driver application in build. Default: True
CHAPTER2_BUILD_DRIVER_APPLICATION:BOOL=TRUE
// Path to a program.
CMAKE_ADDR2LINE:FILEPATH=/usr/bin/addr2line
// Path to a program.
CMAKE_AR:FILEPATH=/usr/bin/ar
// Choose the type of build, options are: None Debug Release RelWithDebInfo MinSizeRel ...
CMAKE_BUILD_TYPE:STRING=
// Enable/Disable color output during build.
CMAKE_COLOR_MAKEFILE:BOOL=ON

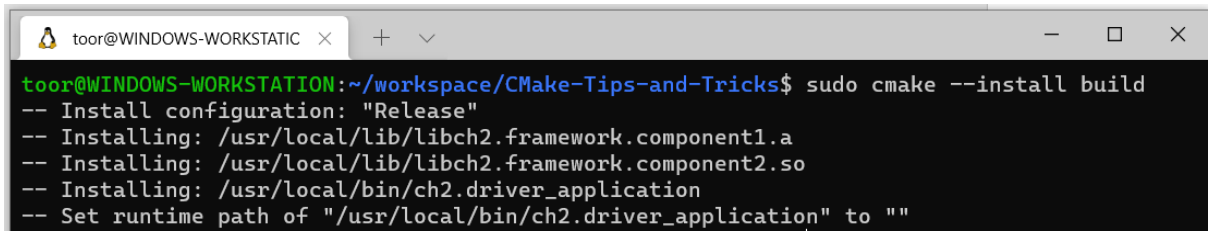

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake --build build/
[50%] Building CXX object chapter_1/CMakeFiles/Chapter1.dir/src/main.cpp.o
[100%] Linking CXX executable Chapter1
[100%] Built target Chapter1
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$

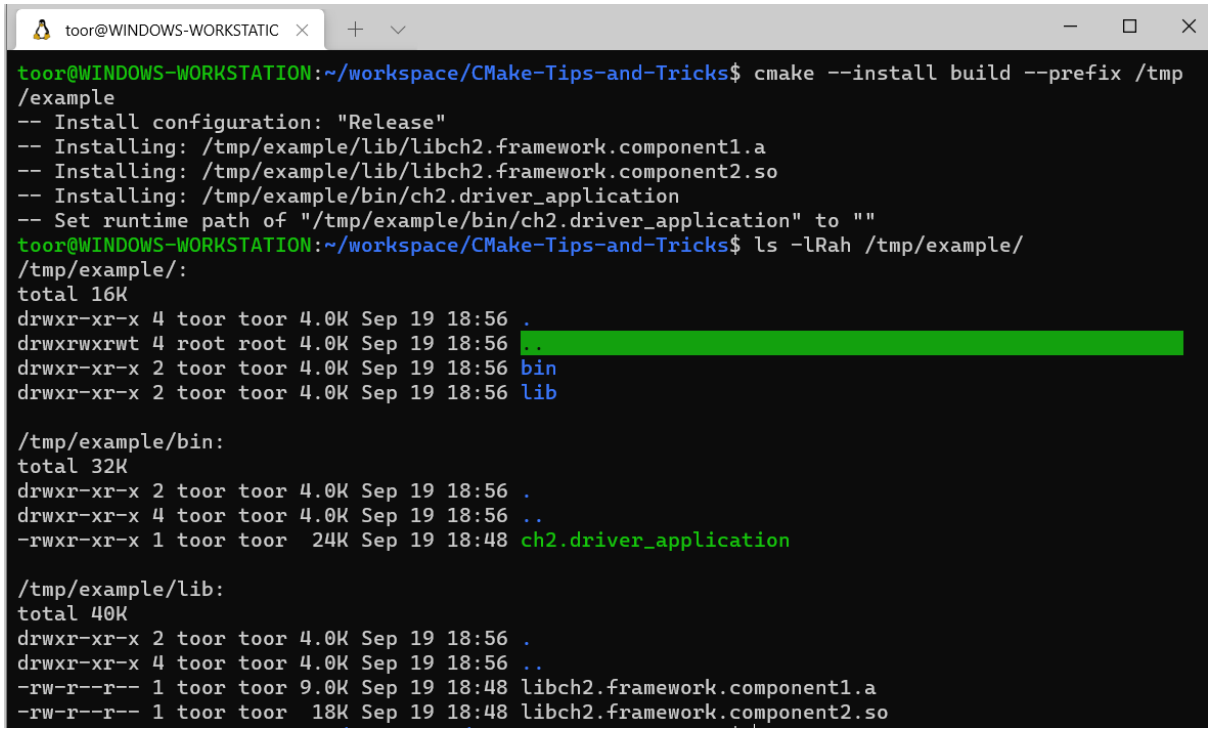

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ /usr/bin/time --format='%C took
%e seconds' cmake --build build --clean-first --parallel 1 1> /dev/null
cmake --build build --clean-first --parallel 1 took 1.26 seconds

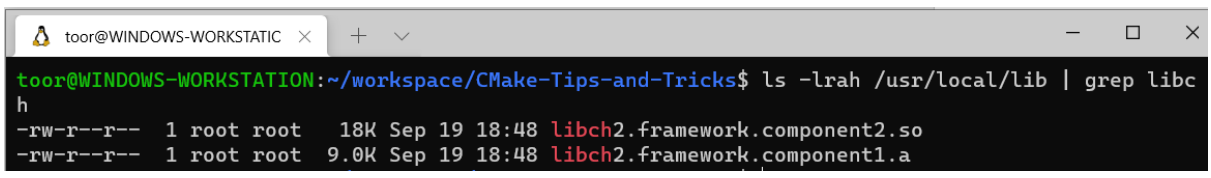

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ /usr/bin/time --format='%C took
%e seconds' cmake --build build --clean-first --parallel 2 1> /dev/null
cmake --build build --clean-first --parallel 2 took 0.96 seconds

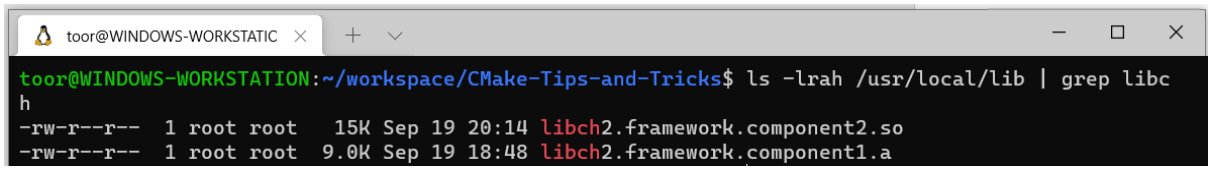

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ /usr/bin/time --format='%C took
%e seconds' cmake --build build --clean-first --parallel 3 1> /dev/null
cmake --build build --clean-first --parallel 3 took 0.76 seconds

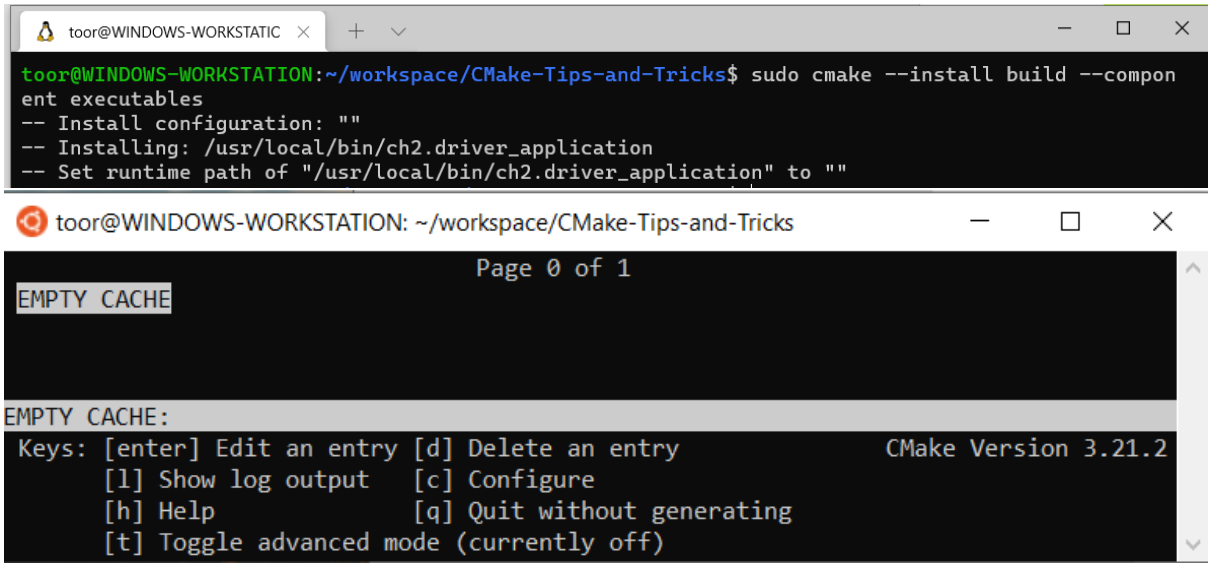

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ /usr/bin/time --format='%C took
%e seconds' cmake --build build --clean-first --parallel 4 1> /dev/null
cmake --build build --clean-first --parallel 4 took 0.75 seconds

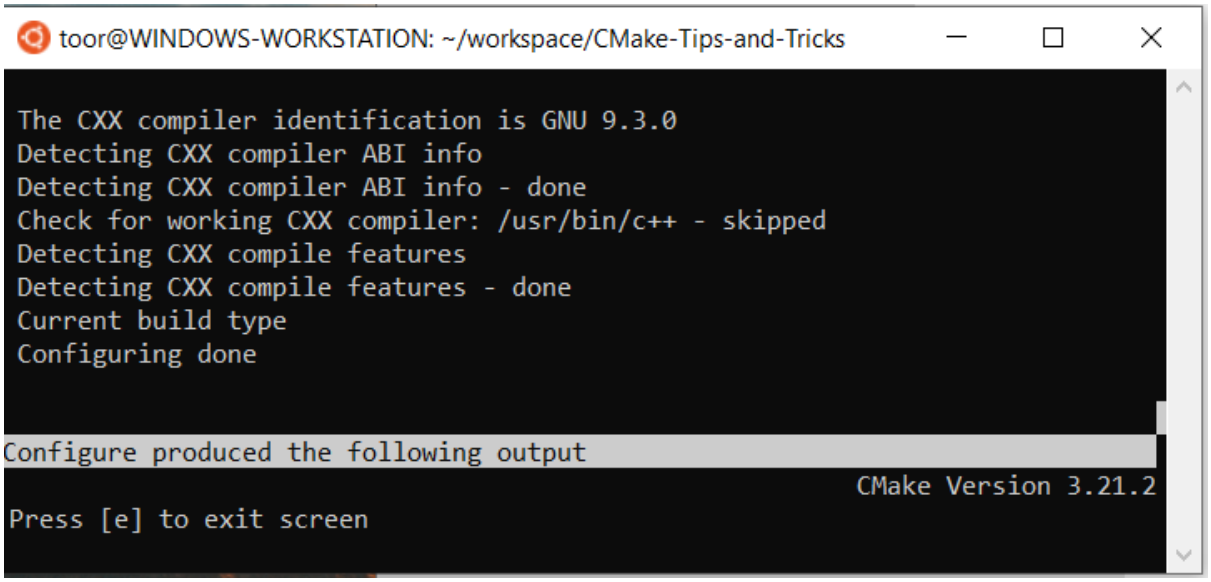

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ sudo cmake --install build
-- Install configuration: "Release"
-- Installing: /usr/local/lib/libch2.framework.component1.a
-- Installing: /usr/local/lib/libch2.framework.component2.so
-- Installing: /usr/local/bin/ch2.driver_application
-- Set runtime path of "/usr/local/bin/ch2.driver_application" to ""

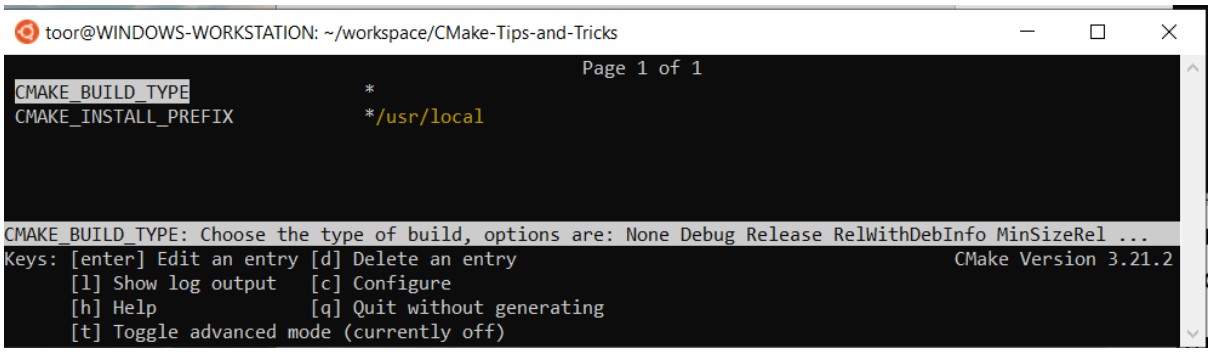

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ cmake --install build --prefix /tmp
/example
-- Install configuration: "Release"
-- Installing: /tmp/example/lib/libch2.framework.component1.a
-- Installing: /tmp/example/lib/libch2.framework.component2.so
-- Installing: /tmp/example/bin/ch2.driver_application
-- Set runtime path of "/tmp/example/bin/ch2.driver_application" to ""
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ ls -lRah /tmp/example/
/tmp/example/:
total 16K
drwxr-xr-x 4 toor toor 4.0K Sep 19 18:56 .
drwxrwxrwt 4 root root 4.0K Sep 19 18:56 ..
drwxr-xr-x 2 toor toor 4.0K Sep 19 18:56 bin
drwxr-xr-x 2 toor toor 4.0K Sep 19 18:56 lib
/tmp/example/bin:
total 32K
drwxr-xr-x 2 toor toor 4.0K Sep 19 18:56 .
drwxr-xr-x 4 toor toor 4.0K Sep 19 18:56 ..
-rwxr-xr-x 1 toor toor 24K Sep 19 18:48 ch2.driver_application
/tmp/example/lib:
total 40K
drwxr-xr-x 2 toor toor 4.0K Sep 19 18:56 .
drwxr-xr-x 4 toor toor 4.0K Sep 19 18:56 ..
-rw-r--r-- 1 toor toor 9.0K Sep 19 18:48 libch2.framework.component1.a
-rw-r--r-- 1 toor toor 18K Sep 19 18:48 libch2.framework.component2.so

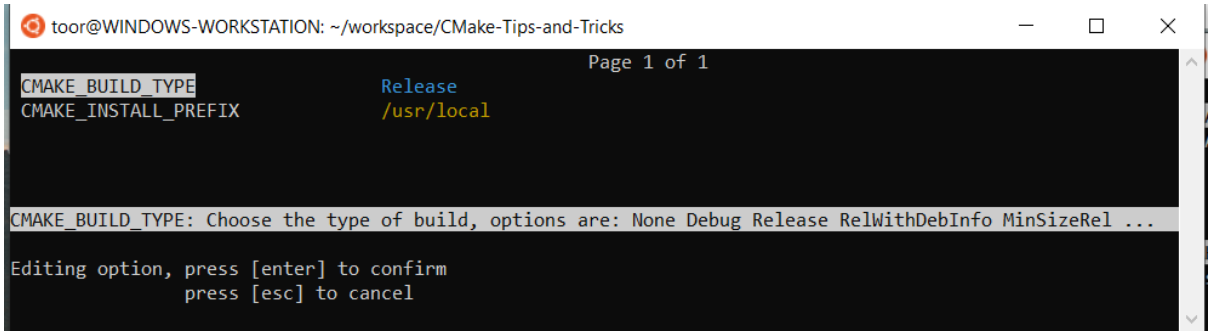

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ ls -lrah /usr/local/lib | grep libc
h
-rw-r--r-- 1 root root 18K Sep 19 18:48 libch2.framework.component2.so
-rw-r--r-- 1 root root 9.0K Sep 19 18:48 libch2.framework.component1.a

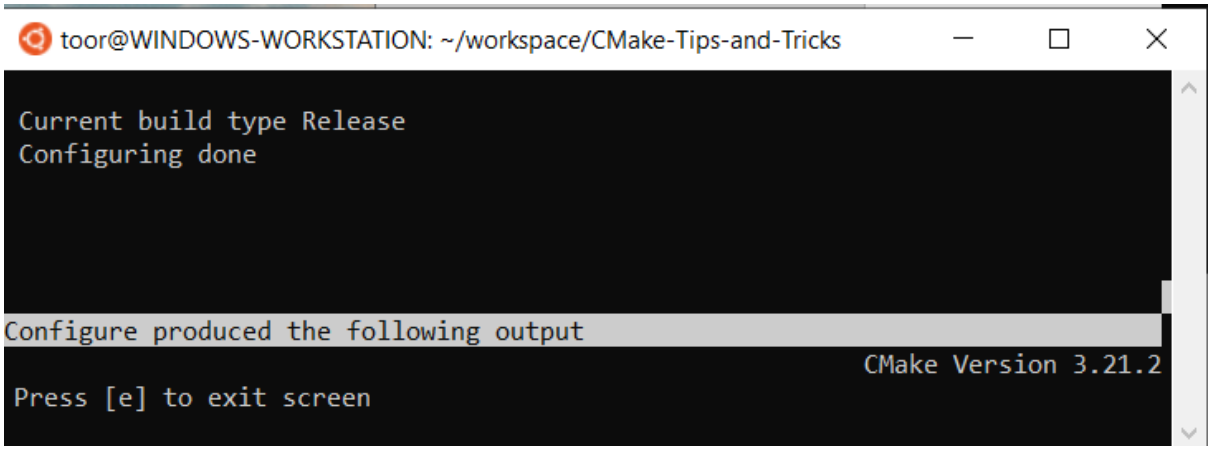

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ ls -lrah /usr/local/lib | grep libc
h
-rw-r--r-- 1 root root 15K Sep 19 20:14 libch2.framework.component2.so
-rw-r--r-- 1 root root 9.0K Sep 19 18:48 libch2.framework.component1.a

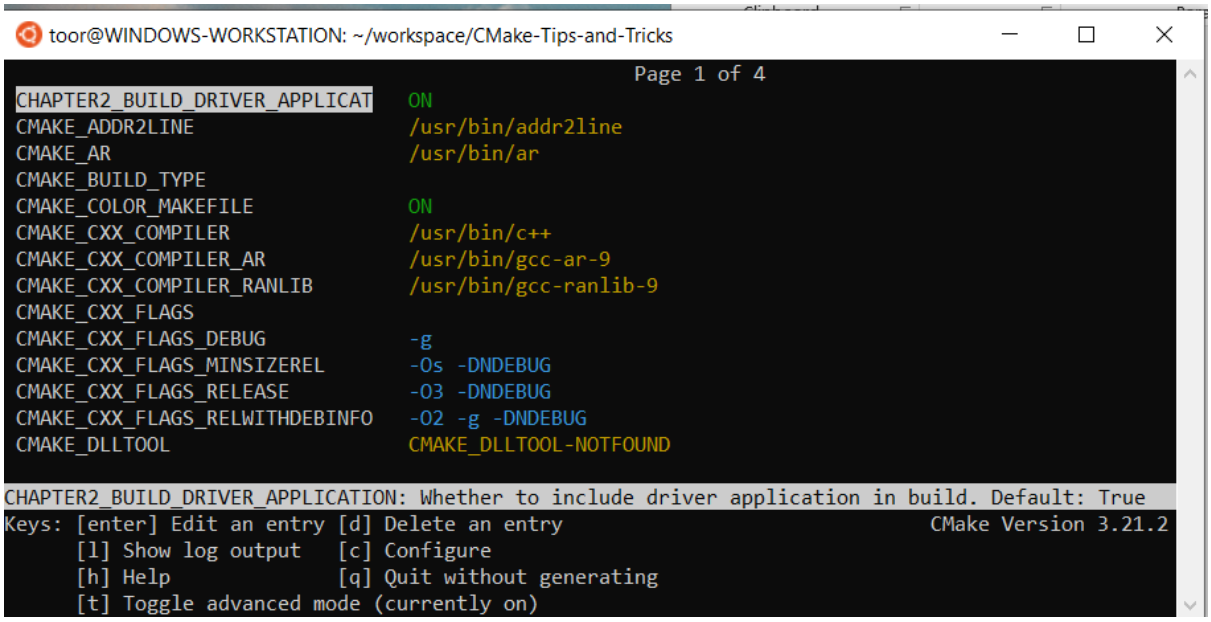

toor@WINDOWS-WORKSTATIC
toor@WINDOWS-WORKSTATION:~/workspace/CMake-Tips-and-Tricks$ sudo cmake --install build --compon
ent executables
-- Install configuration: ""
-- Installing: /usr/local/bin/ch2.driver_application
-- Set runtime path of "/usr/local/bin/ch2.driver_application" to ""
toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
Page 0 of 1
EMPTY CACHE
EMPTY CACHE:
Keys: [enter] Edit an entry [d] Delete an entry
CMake Version 3.21.2
[l] Show log output [c] Configure
[h] Help [q] Quit without generating
[t] Toggle advanced mode (currently off)

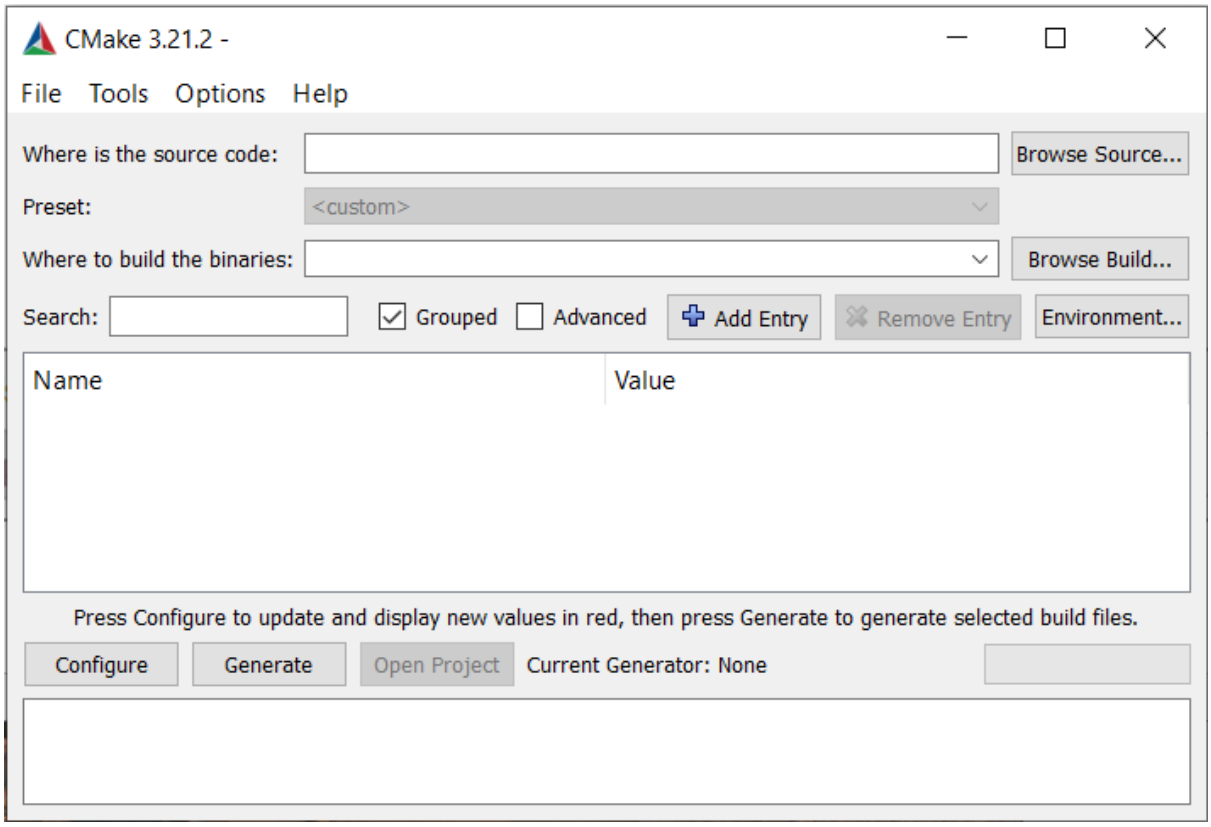

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
The CXX compiler identification is GNU 9.3.0
Detecting CXX compiler ABI info
Detecting CXX compiler ABI info - done
Check for working CXX compiler: /usr/bin/c++ - skipped
Detecting CXX compile features
Detecting CXX compile features - done
Current build type
Configuring done
Configure produced the following output
CMake Version 3.21.2
Press [e] to exit screen

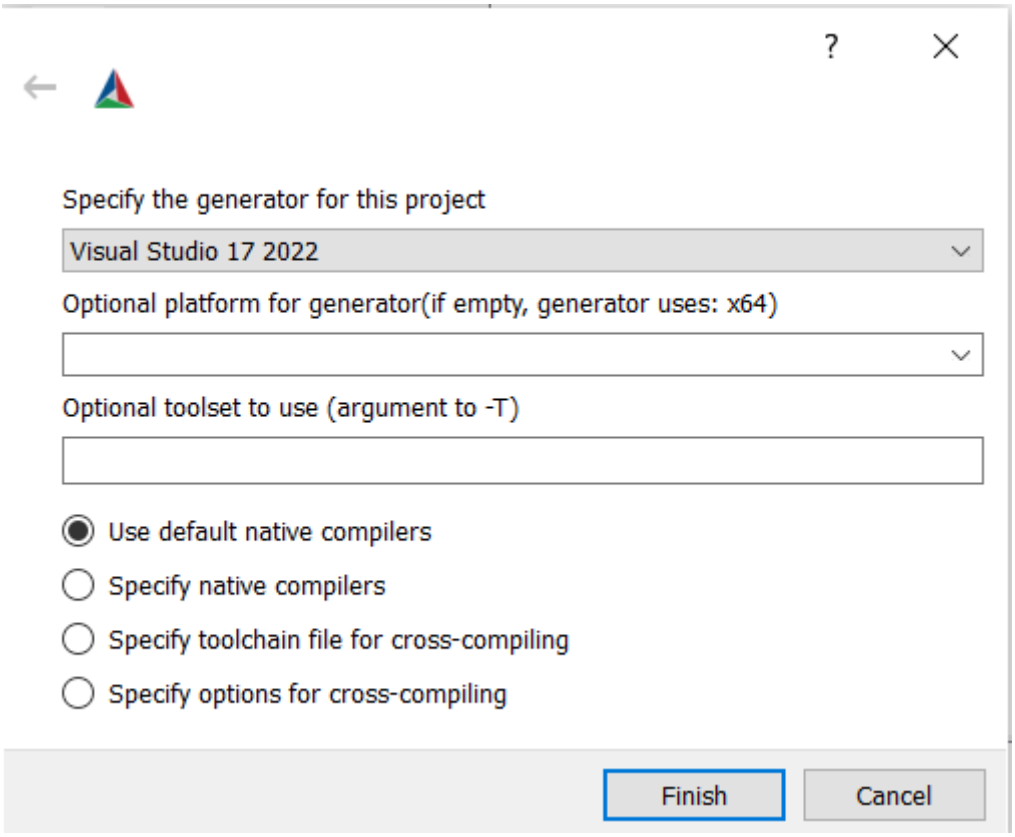

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
Page 1 of 1
CMAKE_BUILD_TYPE *
CMAKE_INSTALL_PREFIX */usr/local
CMAKE_BUILD_TYPE: Choose the type of build, options are: None Debug Release RelWithDebInfo MinSizeRel ...
Keys: [enter] Edit an entry [d] Delete an entry
CMake Version 3.21.2
[l] Show log output [c] Configure
[h] Help [q] Quit without generating
[t] Toggle advanced mode (currently off)

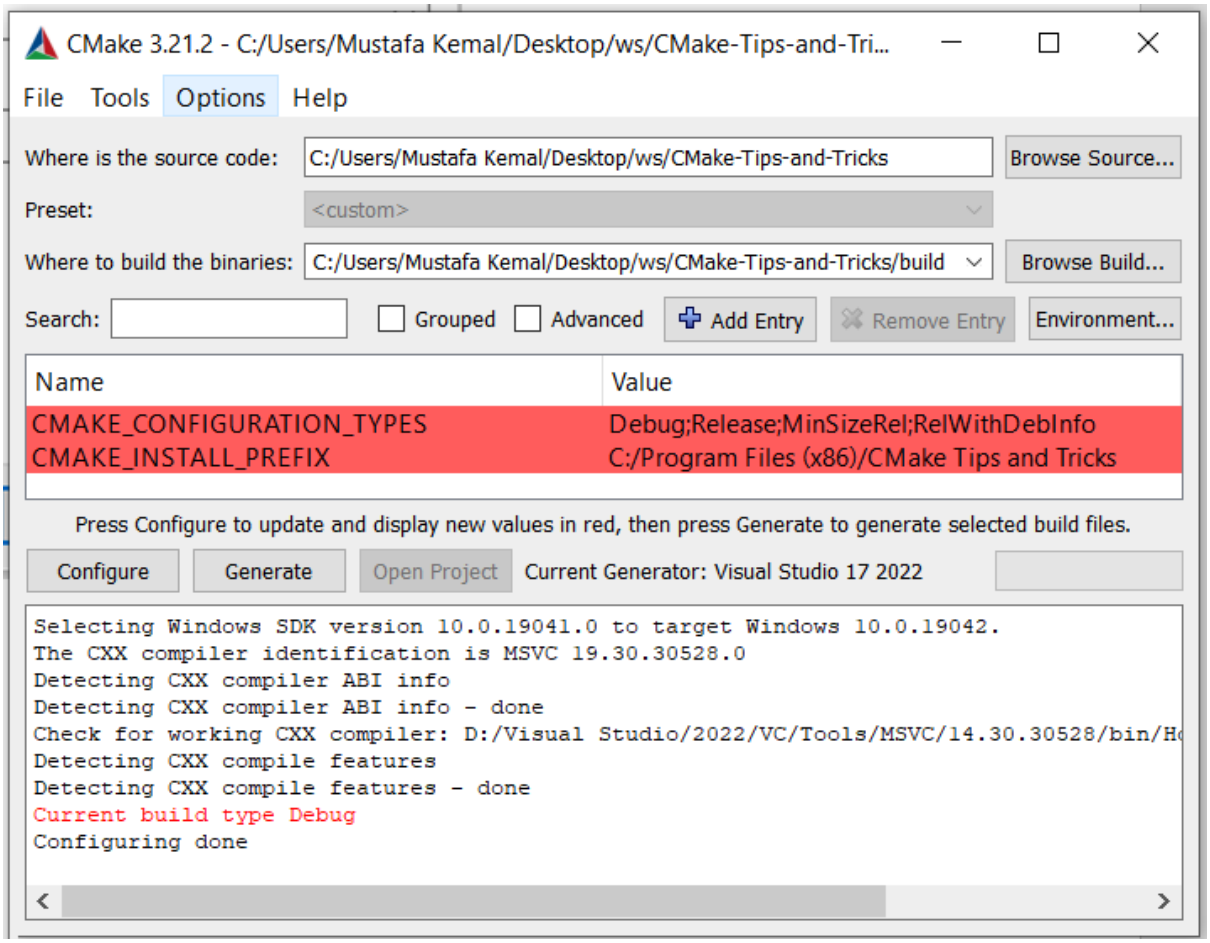

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
Page 1 of 1
CMAKE_BUILD_TYPE Release
CMAKE_INSTALL_PREFIX /usr/local
CMAKE_BUILD_TYPE: Choose the type of build, options are: None Debug Release RelWithDebInfo MinSizeRel ...
Editing option, press [enter] to confirm
press [esc] to cancel

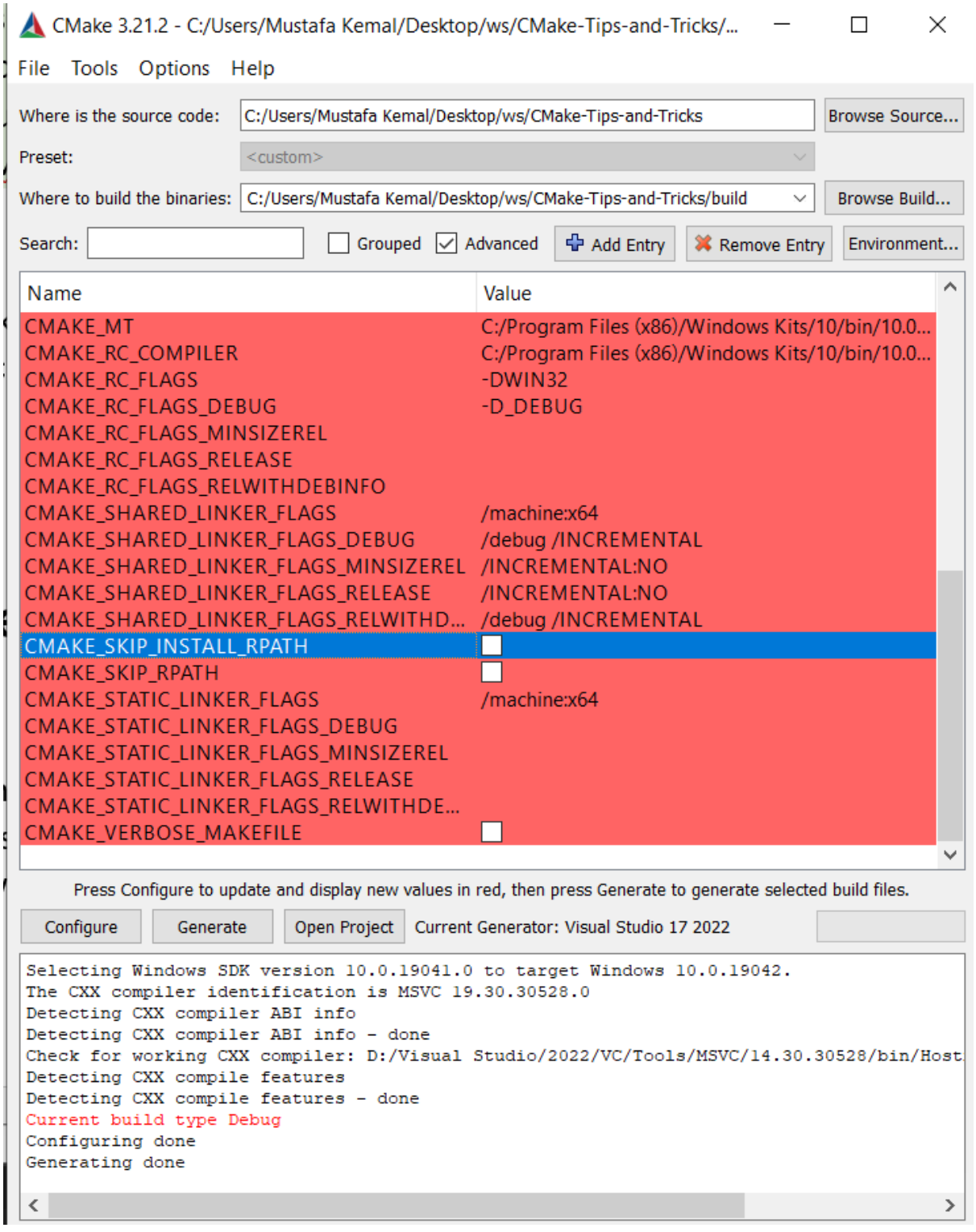

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
Current build type Release
Configuring done
Configure produced the following output
CMake Version 3.21.2
Press [e] to exit screen

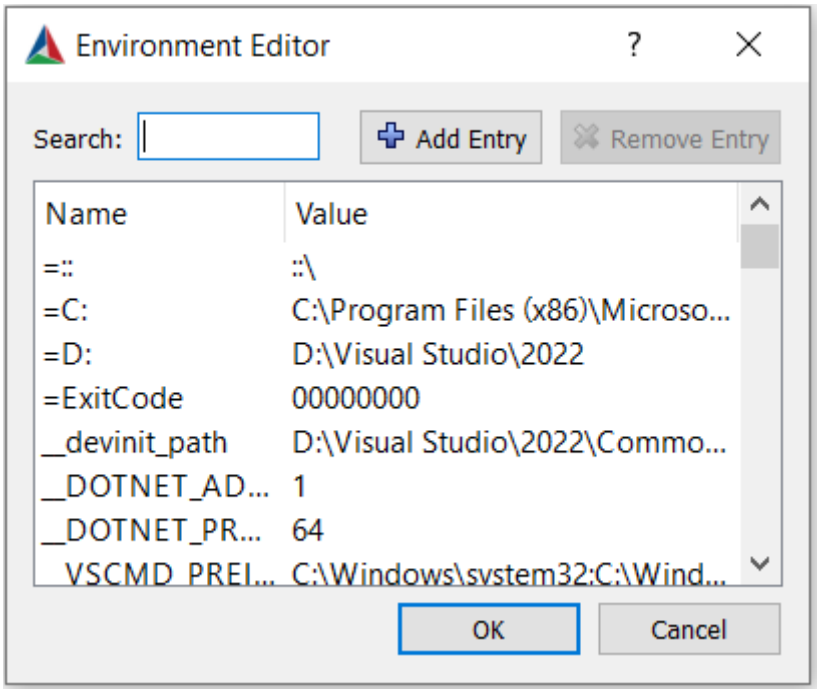

toor@WINDOWS-WORKSTATION: ~/workspace/CMake-Tips-and-Tricks
Page 1 of 4
CHAPTER2_BUILD_DRIVER_APPLICAT ON
CMAKE_ADDR2LINE /usr/bin/addr2line
CMAKE_AR /usr/bin/ar
CMAKE_BUILD_TYPE
CMAKE_COLOR_MAKEFILE ON
CMAKE_CXX_COMPILER /usr/bin/c++
CMAKE_CXX_COMPILER_AR /usr/bin/gcc-ar-9
CMAKE_CXX_COMPILER_RANLIB /usr/bin/gcc-ranlib-9
CMAKE_CXX_FLAGS
CMAKE_CXX_FLAGS_DEBUG -g
CMAKE_CXX_FLAGS_MINSIZEREL -Os -DNDEBUG
CMAKE_CXX_FLAGS_RELEASE -O3 -DNDEBUG
CMAKE_CXX_FLAGS_RELWITHDEBINFO -O2 -g -DNDEBUG
CMAKE_DLLTOOL CMAKE_DLLTOOL-NOTFOUND
CHAPTER2_BUILD_DRIVER_APPLICATION: Whether to include driver application in build. Default: True
Keys: [enter] Edit an entry [d] Delete an entry
[l] Show log output [c] Configure
[h] Help [q] Quit without generating
[t] Toggle advanced mode (currently on)
CMake Version 3.21.2

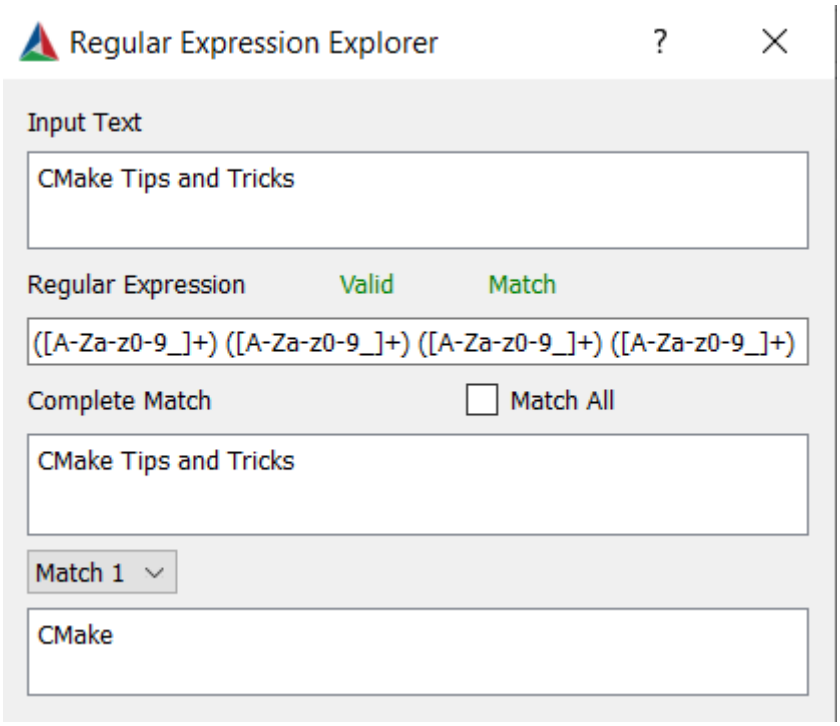

CMake 3.21.2 -
File Tools Options Help
Where is the source code:
Browse Source...
Preset:
<custom>
Where to build the binaries:
Browse Build...
Search:
Grouped
Advanced
Add Entry
Remove Entry
Environment...
Name
Value
Press Configure to update and display new values in red, then press Generate to generate selected build files.
Configure
Generate
Open Project
Current Generator: None

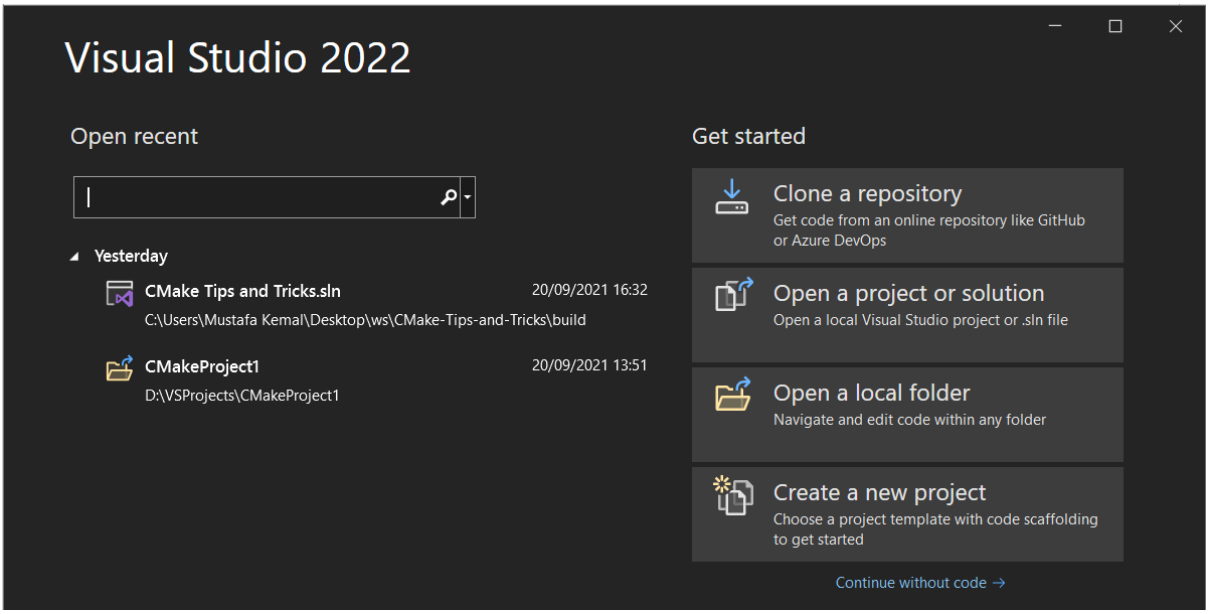

Specify the generator for this project
Visual Studio 17 2022
Optional platform for generator(if empty, generator uses: x64)
Optional toolset to use (argument to -T)
Use default native compilers
Specify native compilers
Specify toolchain file for cross-compiling
Specify options for cross-compiling
Finish
Cancel

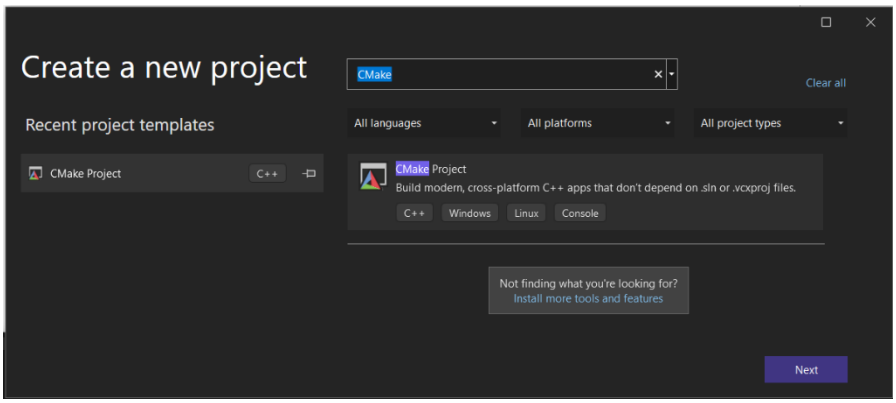

CMake 3.21.2 - C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tri...
File Tools Options Help
Where is the source code: C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tricks
Browse Source...
Preset: <custom>
Where to build the binaries: C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tricks/build
Browse Build...
Search:
Grouped
Advanced
Add Entry
Remove Entry
Environment...
Name
Value
CMAKE_CONFIGURATION_TYPES
Debug;Release;MinSizeRel;RelWithDebInfo
CMAKE_INSTALL_PREFIX
C:/Program Files (x86)/CMake Tips and Tricks
Press Configure to update and display new values in red, then press Generate to generate selected build files.
Configure
Generate
Open Project
Current Generator: Visual Studio 17 2022
Selecting Windows SDK version 10.0.19041.0 to target Windows 10.0.19042.
The CXX compiler identification is MSVC 19.30.30528.0
Detecting CXX compiler ABI info
Detecting CXX compiler ABI info - done
Check for working CXX compiler: D:/Visual Studio/2022/VC/Tools/MSVC/14.30.30528/bin/H
Detecting CXX compile features
Detecting CXX compile features - done
Current build type Debug
Configuring done

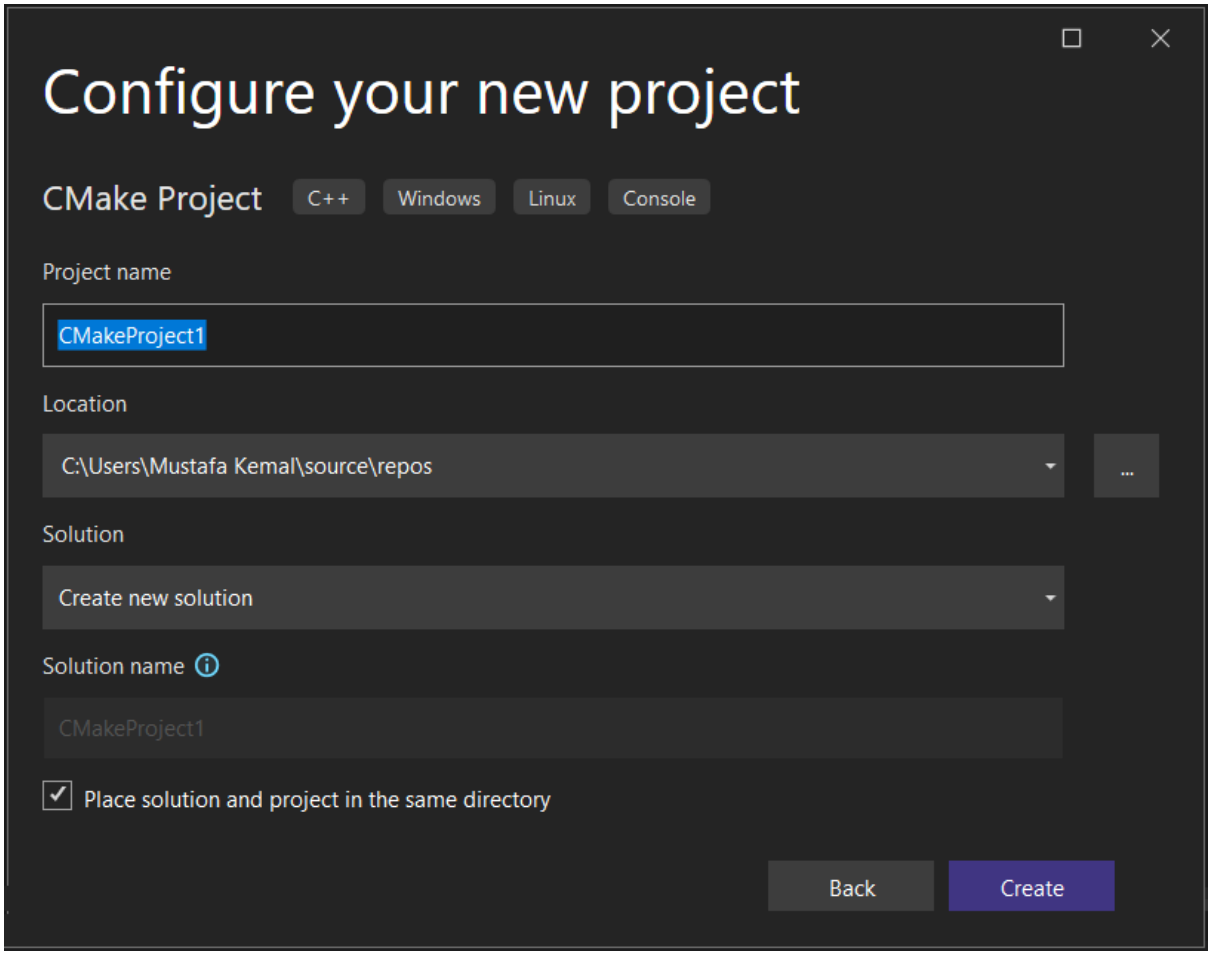

CMake 3.21.2 - C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tricks/...
File Tools Options Help
Where is the source code: C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tricks
Browse Source...
Preset: <custom>
Where to build the binaries: C:/Users/Mustafa Kemal/Desktop/ws/CMake-Tips-and-Tricks/build
Browse Build...
Search:
Grouped
Advanced
Add Entry
Remove Entry
Environment...
Name
Value
CMAKE_MT C:/Program Files (x86)/Windows Kits/10/bin/10.0...
CMAKE_RC_COMPILER C:/Program Files (x86)/Windows Kits/10/bin/10.0...
CMAKE_RC_FLAGS -DWIN32
CMAKE_RC_FLAGS_DEBUG -D_DEBUG
CMAKE_RC_FLAGS_MINSIZEREL
CMAKE_RC_FLAGS_RELEASE
CMAKE_RC_FLAGS_RELWITHDEBINFO
CMAKE_SHARED_LINKER_FLAGS /machine:x64
CMAKE_SHARED_LINKER_FLAGS_DEBUG /debug /INCREMENTAL
CMAKE_SHARED_LINKER_FLAGS_MINSIZEREL /INCREMENTAL:NO
CMAKE_SHARED_LINKER_FLAGS_RELEASE /INCREMENTAL:NO
CMAKE_SHARED_LINKER_FLAGS_RELWITHD... /debug /INCREMENTAL
CMAKE_SKIP_INSTALL_RPATH
CMAKE_SKIP_RPATH
CMAKE_STATIC_LINKER_FLAGS /machine:x64
CMAKE_STATIC_LINKER_FLAGS_DEBUG
CMAKE_STATIC_LINKER_FLAGS_MINSIZEREL
CMAKE_STATIC_LINKER_FLAGS_RELEASE
CMAKE_STATIC_LINKER_FLAGS_RELWITHDE...
CMAKE_VERBOSE_MAKEFILE
Press Configure to update and display new values in red, then press Generate to generate selected build files.
Configure
Generate
Open Project
Current Generator: Visual Studio 17 2022
Selecting Windows SDK version 10.0.19041.0 to target Windows 10.0.19042.
The CXX compiler identification is MSVC 19.30.30528.0
Detecting CXX compiler ABI info
Detecting CXX compiler ABI info - done
Check for working CXX compiler: D:/Visual Studio/2022/VC/Tools/MSVC/14.30.30528/bin/Host
Detecting CXX compile features
Detecting CXX compile features - done
Current build type Debug
Configuring done
Generating done

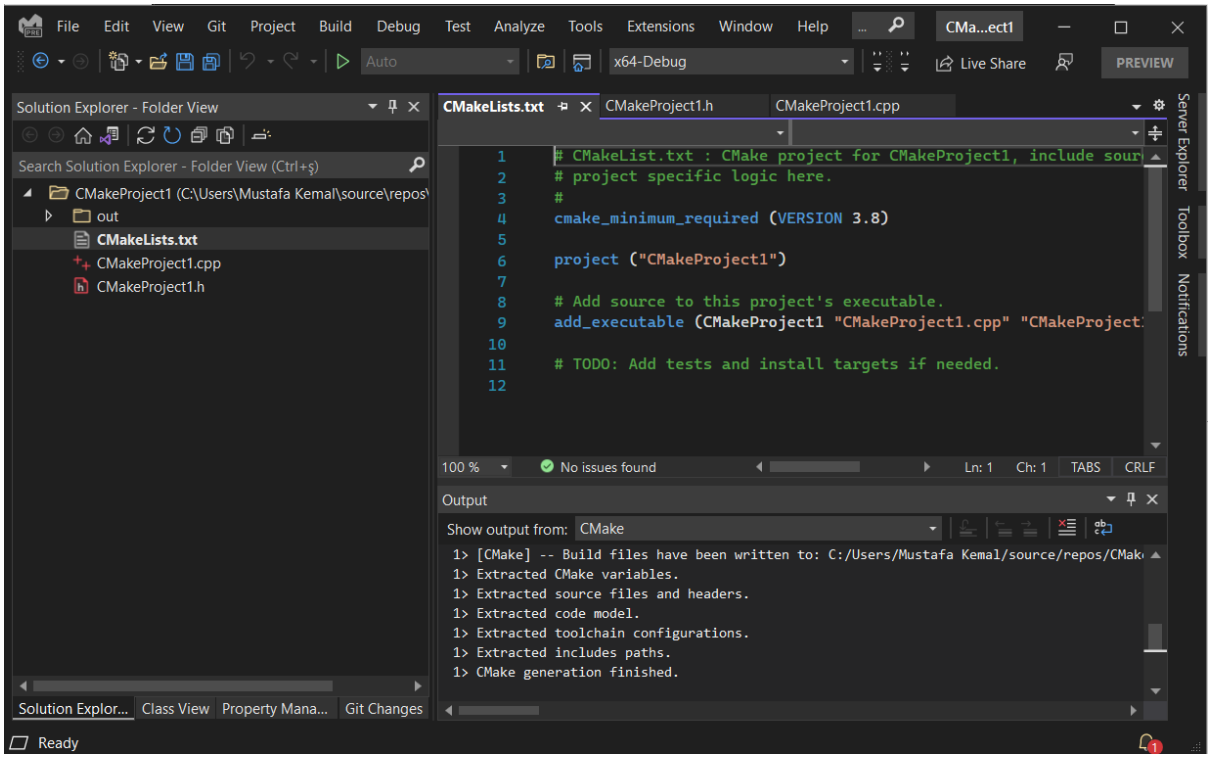

Environment Editor
Search:
Add Entry
Remove Entry
Name
Value
=::
::\
=C:
C:\Program Files (x86)\Microso...
=D:
D:\Visual Studio\2022
=ExitCode
00000000
__devinit_path
D:\Visual Studio\2022\Commo...
__DOTNET_AD...
1
__DOTNET_PR...
64
VSCMD_PREI...
C:\Windows\system32;C:\Wind...
OK
Cancel

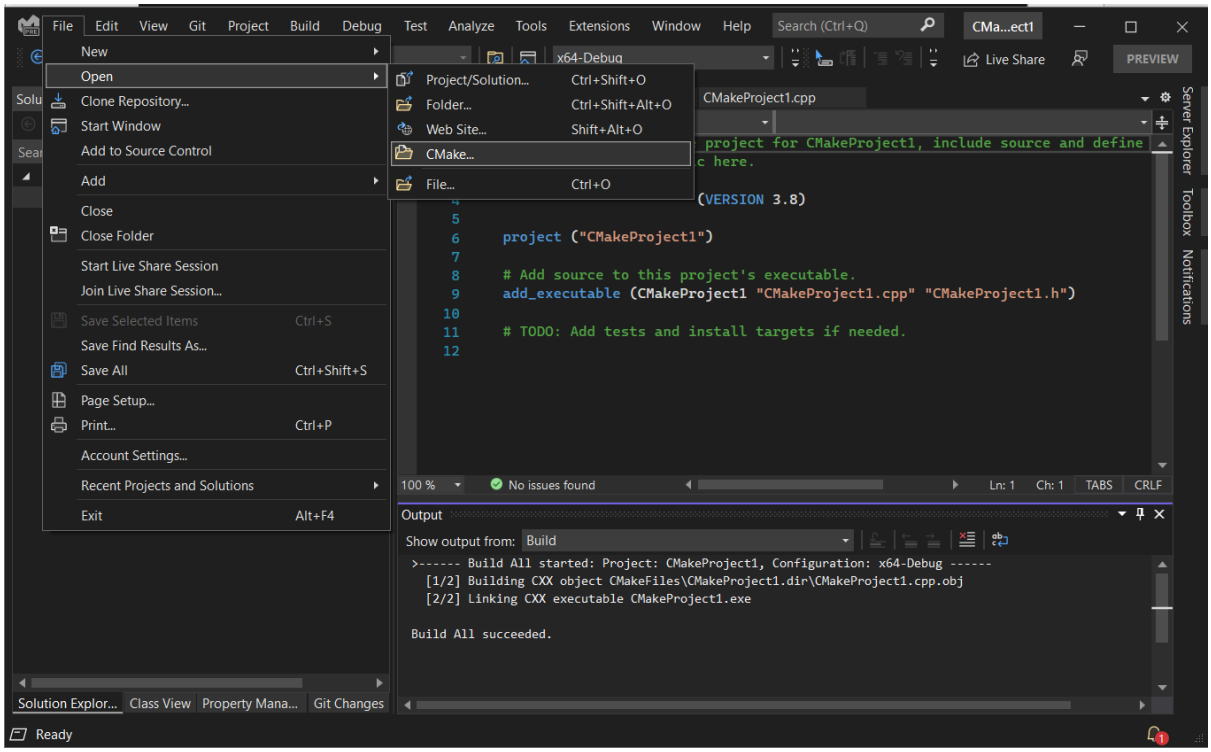

Regular Expression Explorer
Input Text
CMake Tips and Tricks
Regular Expression
Valid
Match
([A-Za-z0-9_]+) ([A-Za-z0-9_]+) ([A-Za-z0-9_]+) ([A-Za-z0-9_]+)
Complete Match
Match All
CMake Tips and Tricks
Match 1
CMake

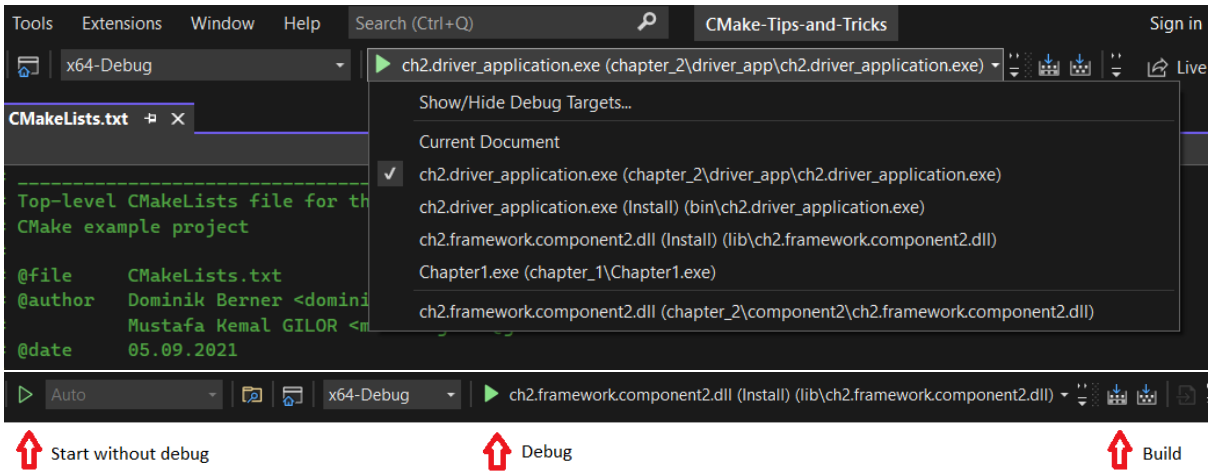

Visual Studio 2022
Open recent
Yesterday
CMake Tips and Tricks.sln
20/09/2021 16:32
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\build
CMakeProject1
20/09/2021 13:51
D:\VSProjects\CMakeProject1
Get started
Clone a repository
Get code from an online repository like GitHub or Azure DevOps
Open a project or solution
Open a local Visual Studio project or .sln file
Open a local folder
Navigate and edit code within any folder
Create a new project
Choose a project template with code scaffolding to get started
Continue without code →

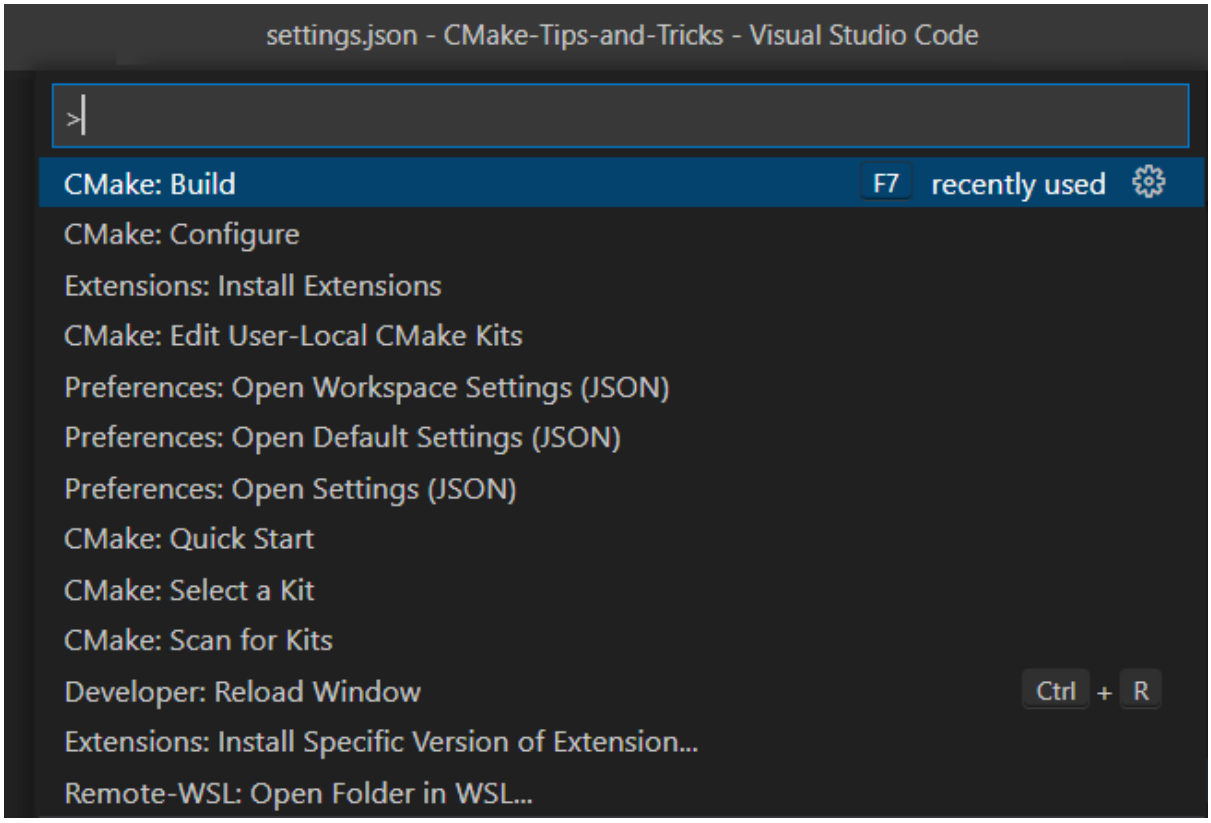

Create a new project
CMake
Clear all
Recent project templates
All languages
All platforms
All project types
CMake Project
C++
CMake Project
Build modern, cross-platform C++ apps that don't depend on .sln or .vcxproj files.
C++
Windows
Linux
Console
Not finding what you're looking for?
Install more tools and features
Next

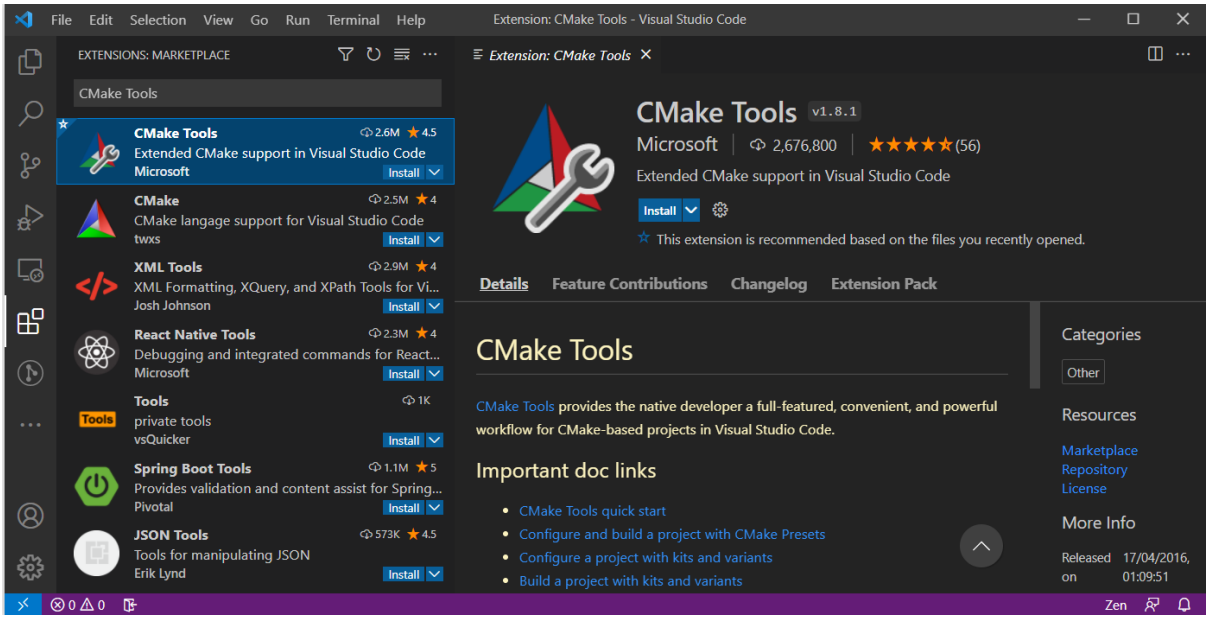

Configure your new project
CMake Project
C++
Windows
Linux
Console
Project name
CMakeProject1
Location
C:\Users\Mustafa Kemal\source\repos
...
Solution
Create new solution
Solution name
CMakeProject1
Place solution and project in the same directory
Back
Create

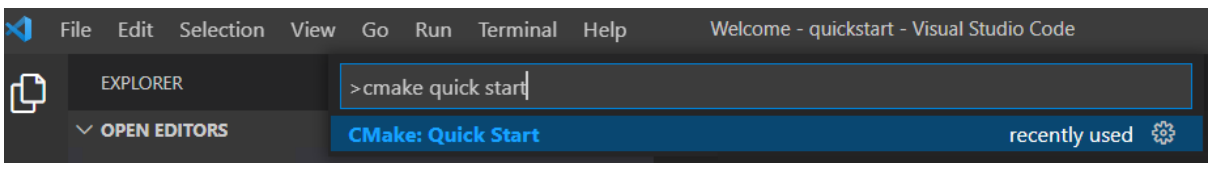

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
CMa...ect1
Auto
x64-Debug
Live Share
PREVIEW
Solution Explorer - Folder View
Search Solution Explorer - Folder View (Ctrl+ş)
CMakeProject1 (C:\Users\Mustafa Kemal\source\repos\
out
CMakeLists.txt
CMakeProject1.cpp
CMakeProject1.h
CMakeLists.txt
CMakeProject1.h
CMakeProject1.cpp
CMakeList.txt : CMake project for CMakeProject1, include sour
project specific logic here.
#
cmake_minimum_required (VERSION 3.8)
project ("CMakeProject1")
Add source to this project's executable.
add_executable (CMakeProject1 "CMakeProject1.cpp" "CMakeProject
TODO: Add tests and install targets if needed.
100 %
No issues found
Ln: 1
Ch: 1
TABS
CRLF
Output
Show output from: CMake
1> [CMake] -- Build files have been written to: C:/Users/Mustafa Kemal/source/repos/CMak
1> Extracted CMake variables.
1> Extracted source files and headers.
1> Extracted code model.
1> Extracted toolchain configurations.
1> Extracted includes paths.
1> CMake generation finished.
Solution Explor...
Class View
Property Mana...
Git Changes
Server Explorer
Toolbox
Notifications
Ready

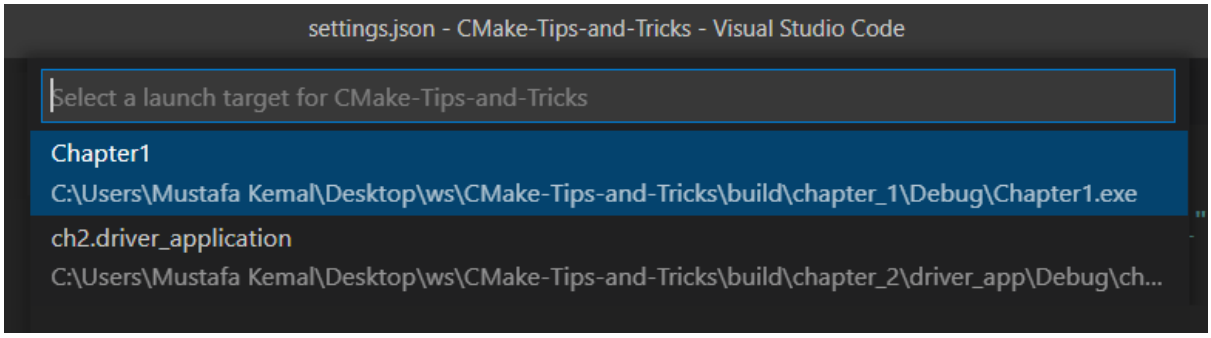

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
New
Open
Clone Repository...
Start Window
Add to Source Control
Add
Close
Close Folder
Start Live Share Session
Join Live Share Session...
Save Selected Items Ctrl+S
Save Find Results As...
Save All Ctrl+Shift+S
Page Setup...
Print... Ctrl+P
Account Settings...
Recent Projects and Solutions
Exit Alt+F4
Project/Solution... Ctrl+Shift+O
Folder... Ctrl+Shift+Alt+O
Web Site... Shift+Alt+O
CMake...
File... Ctrl+O
x64-Debug
Live Share
PREVIEW
CMakeProject1.cpp
project for CMakeProject1, include source and define
c here.
(VERSION 3.8)
project ("CMakeProject1")
Add source to this project's executable.
add_executable (CMakeProject1 "CMakeProject1.cpp" "CMakeProject1.h")
TODO: Add tests and install targets if needed.
100 % No issues found Ln: 1 Ch: 1 TABS CRLF
Output
Show output from: Build
>------ Build All started: Project: CMakeProject1, Configuration: x64-Debug ------
[1/2] Building CXX object CMakeFiles\CMakeProject1.dir\CMakeProject1.cpp.obj
[2/2] Linking CXX executable CMakeProject1.exe
Build All succeeded.
Solution Explor... Class View Property Mana... Git Changes
Ready

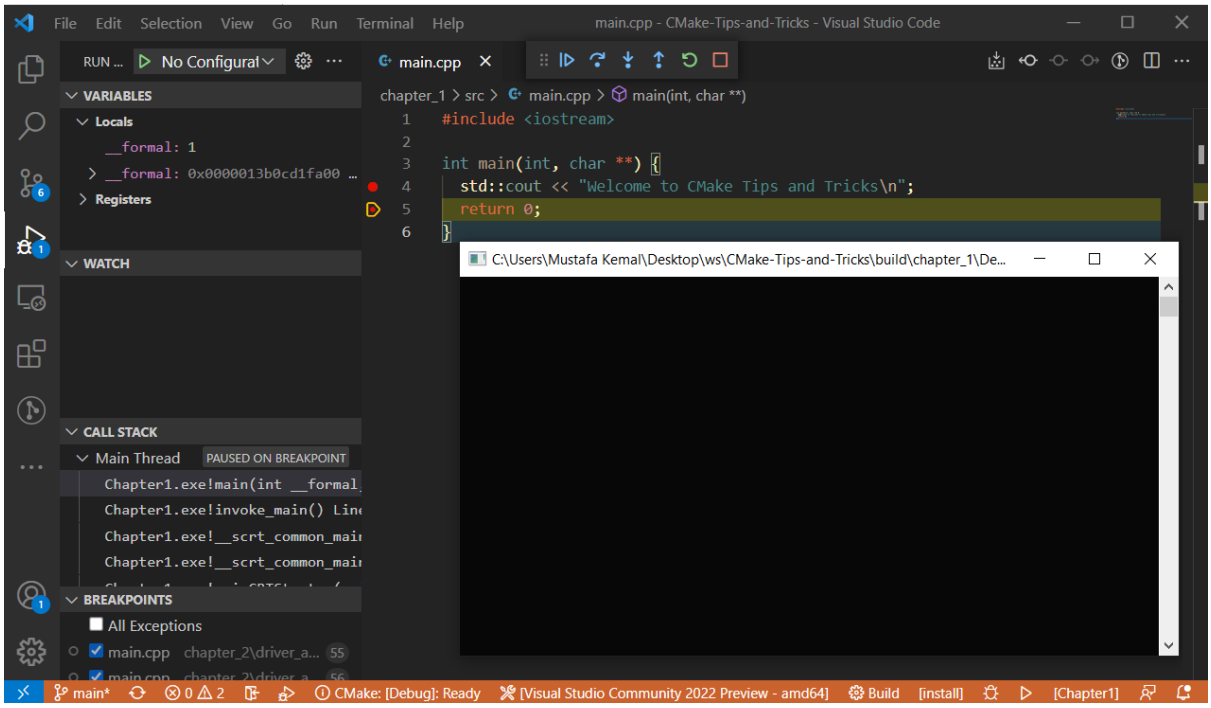

Tools Extensions Window Help
Search (Ctrl+Q)
CMake-Tips-and-Tricks
Sign in
x64-Debug
ch2.driver_application.exe (chapter_2\driver_app\ch2.driver_application.exe)
Show/Hide Debug Targets...
Current Document
ch2.driver_application.exe (chapter_2\driver_app\ch2.driver_application.exe)
ch2.driver_application.exe (Install) (bin\ch2.driver_application.exe)
ch2.framework.component2.dll (Install) (lib\ch2.framework.component2.dll)
Chapter1.exe (chapter_1\Chapter1.exe)
ch2.framework.component2.dll (chapter_2\component2\ch2.framework.component2.dll)
CMakeLists.txt
Top-level CMakeLists file for th
CMake example project
@file CMakeLists.txt
@author Dominik Berner <domini
Mustafa Kemal GILOR <m
@date 05.09.2021
Auto
x64-Debug
ch2.framework.component2.dll (Install) (lib\ch2.framework.component2.dll)
Start without debug
Debug
Build

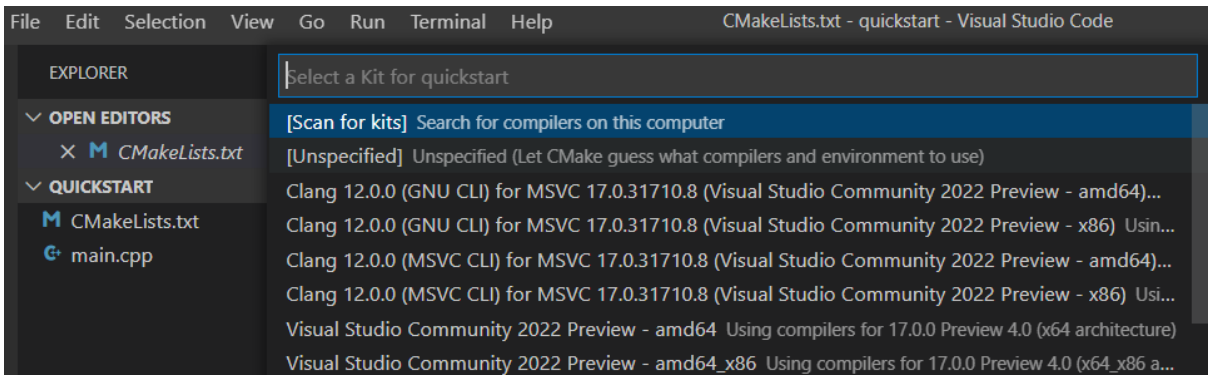

settings.json - CMake-Tips-and-Tricks - Visual Studio Code
>
CMake: Build
F7
recently used
CMake: Configure
Extensions: Install Extensions
CMake: Edit User-Local CMake Kits
Preferences: Open Workspace Settings (JSON)
Preferences: Open Default Settings (JSON)
Preferences: Open Settings (JSON)
CMake: Quick Start
CMake: Select a Kit
CMake: Scan for Kits
Developer: Reload Window
Ctrl + R
Extensions: Install Specific Version of Extension...
Remote-WSL: Open Folder in WSL...

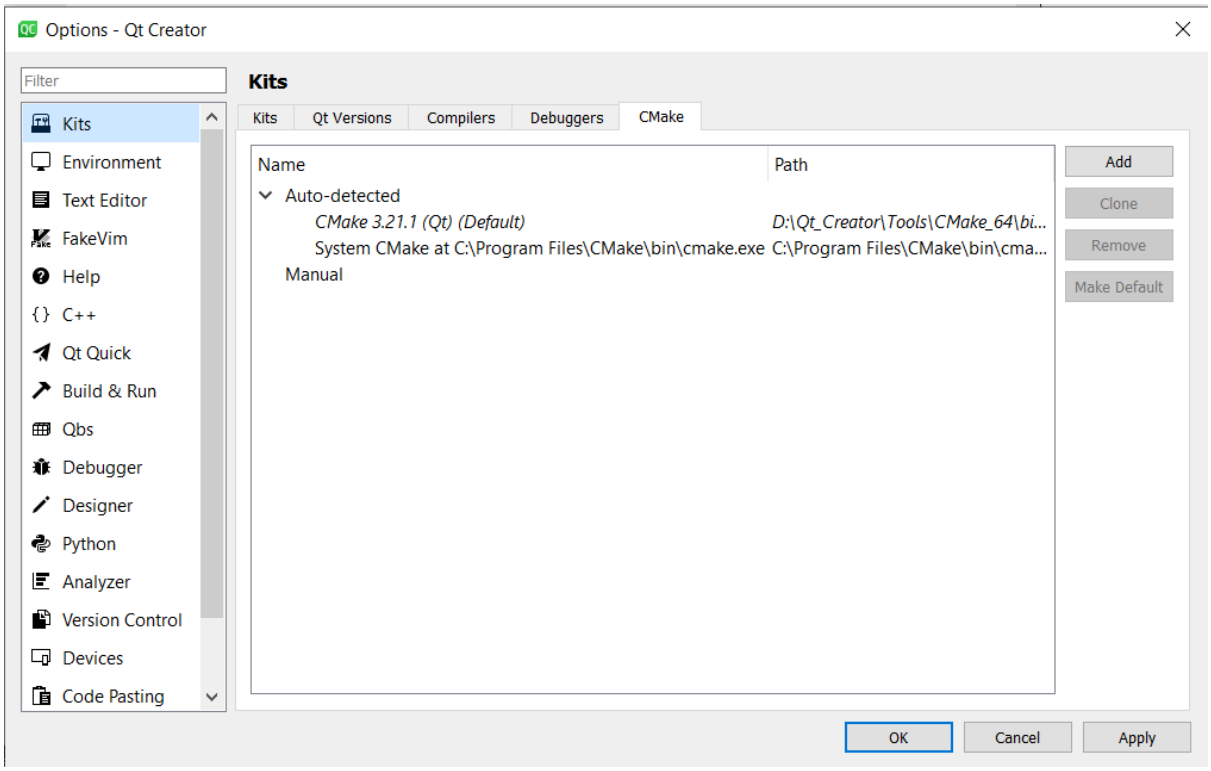

File Edit Selection View Go Run Terminal Help
Extension: CMake Tools - Visual Studio Code
EXTENSIONS: MARKETPLACE
CMake Tools
CMake Tools 2.6M 4.5
Extended CMake support in Visual Studio Code
Microsoft
Install
CMake 2.5M 4
CMake langage support for Visual Studio Code
twxs
XML Tools 2.9M 4
XML Formatting, XQuery, and XPath Tools for Vi...
Josh Johnson
React Native Tools 2.3M 4
Debugging and integrated commands for React...
Microsoft
Tools 1K
private tools
vsQuicker
Spring Boot Tools 1.1M 5
Provides validation and content assist for Spring...
Pivotal
JSON Tools 573K 4.5
Tools for manipulating JSON
Erik Lynd
Extension: CMake Tools
CMake Tools v1.8.1
Microsoft 2,676,800 (56)
Extended CMake support in Visual Studio Code
Install
This extension is recommended based on the files you recently opened.
Details
Feature Contributions
Changelog
Extension Pack
CMake Tools
CMake Tools provides the native developer a full-featured, convenient, and powerful workflow for CMake-based projects in Visual Studio Code.
Important doc links
CMake Tools quick start
Configure and build a project with CMake Presets
Configure a project with kits and variants
Build a project with kits and variants
Categories
Other
Resources
Marketplace
Repository
License
More Info
Released on 17/04/2016, 01:09:51
Zen

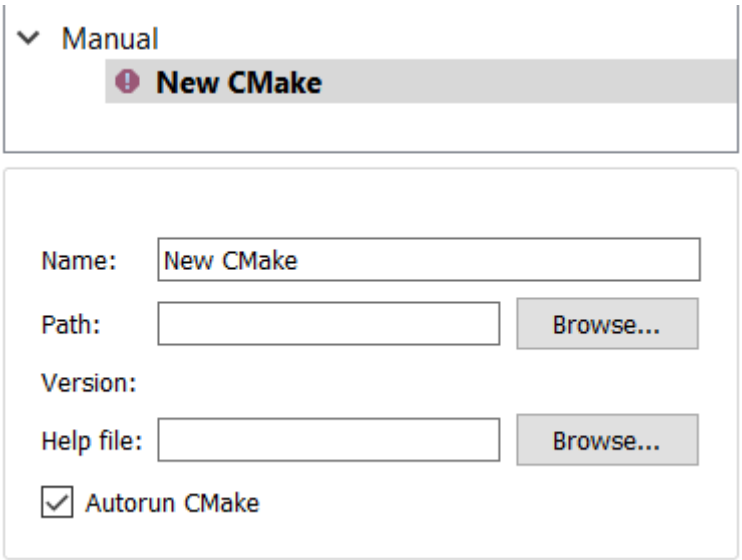

File Edit Selection View Go Run Terminal Help
Welcome - quickstart - Visual Studio Code
EXPLORER
OPEN EDITORS
>cmake quick start
CMake: Quick Start
recently used

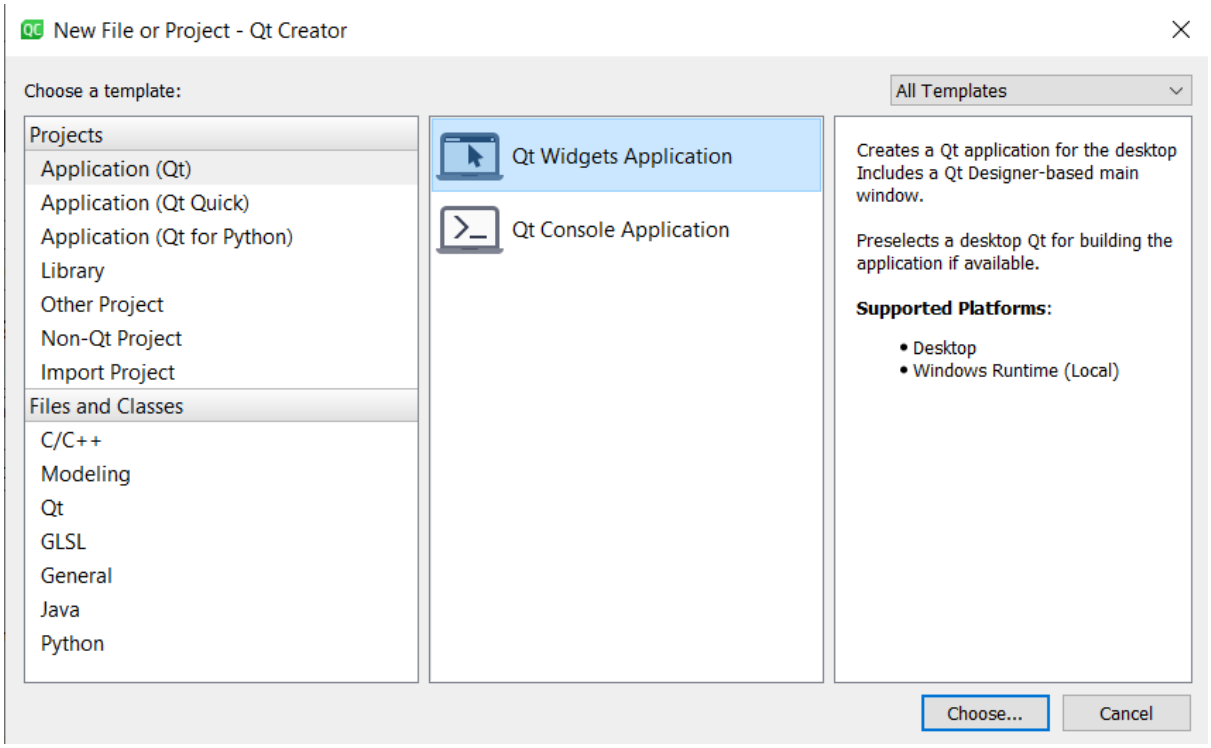

settings.json - CMake-Tips-and-Tricks - Visual Studio Code
Select a launch target for CMake-Tips-and-Tricks
Chapter1
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\build\chapter_1\Debug\Chapter1.exe
ch2.driver_application
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\build\chapter_2\driver_app\Debug\ch...

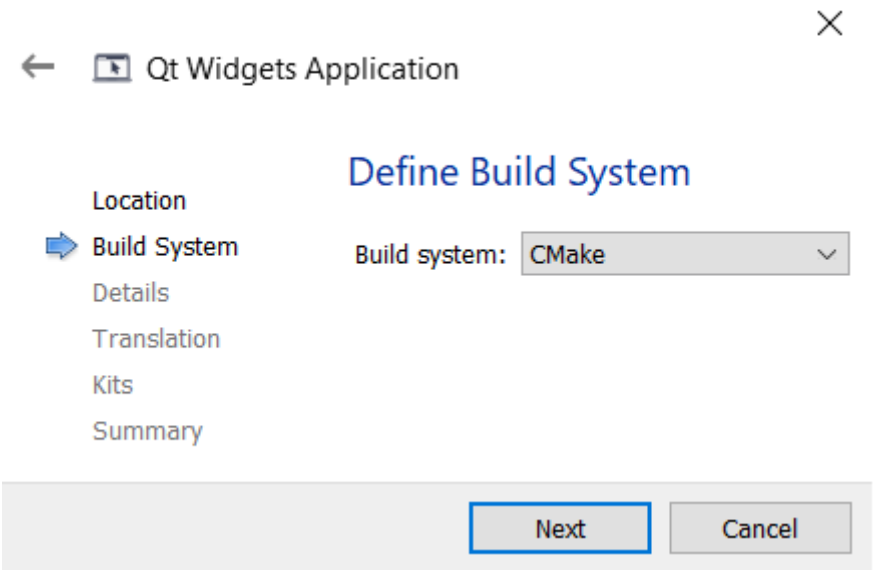

File Edit Selection View Go Run Terminal Help
main.cpp - CMake-Tips-and-Tricks - Visual Studio Code
RUN ... No Configurat
main.cpp
VARIABLES
Locals
__formal: 1
__formal: 0x0000013b0cd1fa00 ...
Registers
WATCH
CALL STACK
Main Thread PAUSED ON BREAKPOINT
Chapter1.exe!main(int __formal
Chapter1.exe!invoke_main() Lin
Chapter1.exe!__scrt_common_mai
Chapter1.exe!__scrt_common_mai
BREAKPOINTS
All Exceptions
main.cpp chapter_2\driver_a... 55
chapter_1 > src > main.cpp > main(int, char **)
1 #include <iostream>
2
3 int main(int, char **) {
4 std::cout << "Welcome to CMake Tips and Tricks\n";
5 return 0;
6 }
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\build\chapter_1\De...
main* 0 2 CMake: [Debug]: Ready [Visual Studio Community 2022 Preview - amd64] Build [install] [Chapter1]

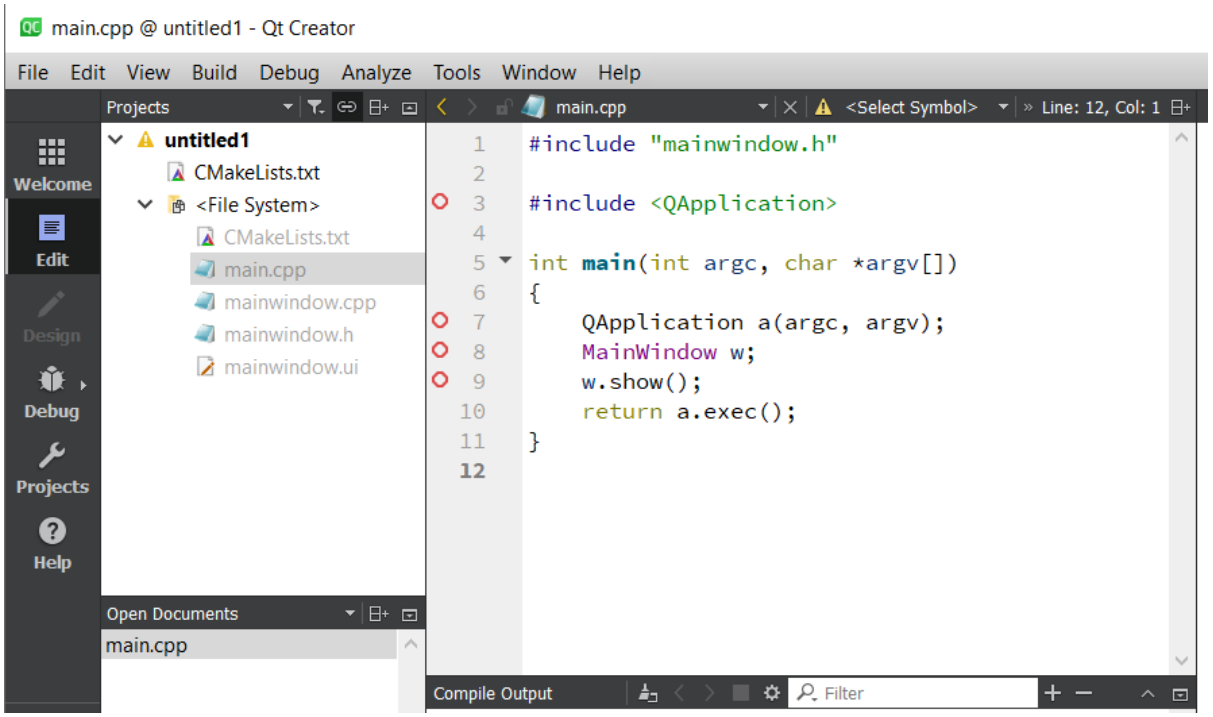

File Edit Selection View Go Run Terminal Help
CMakeLists.txt - quickstart - Visual Studio Code
EXPLORER
OPEN EDITORS
CMakeLists.txt
QUICKSTART
CMakeLists.txt
main.cpp
Select a Kit for quickstart
[Scan for kits] Search for compilers on this computer
[Unspecified] Unspecified (Let CMake guess what compilers and environment to use)
Clang 12.0.0 (GNU CLI) for MSVC 17.0.31710.8 (Visual Studio Community 2022 Preview - amd64)...
Clang 12.0.0 (GNU CLI) for MSVC 17.0.31710.8 (Visual Studio Community 2022 Preview - x86) Usin...
Clang 12.0.0 (MSVC CLI) for MSVC 17.0.31710.8 (Visual Studio Community 2022 Preview - amd64)...
Clang 12.0.0 (MSVC CLI) for MSVC 17.0.31710.8 (Visual Studio Community 2022 Preview - x86) Usi...
Visual Studio Community 2022 Preview - amd64 Using compilers for 17.0.0 Preview 4.0 (x64 architecture)
Visual Studio Community 2022 Preview - amd64_x86 Using compilers for 17.0.0 Preview 4.0 (x64_x86 a...

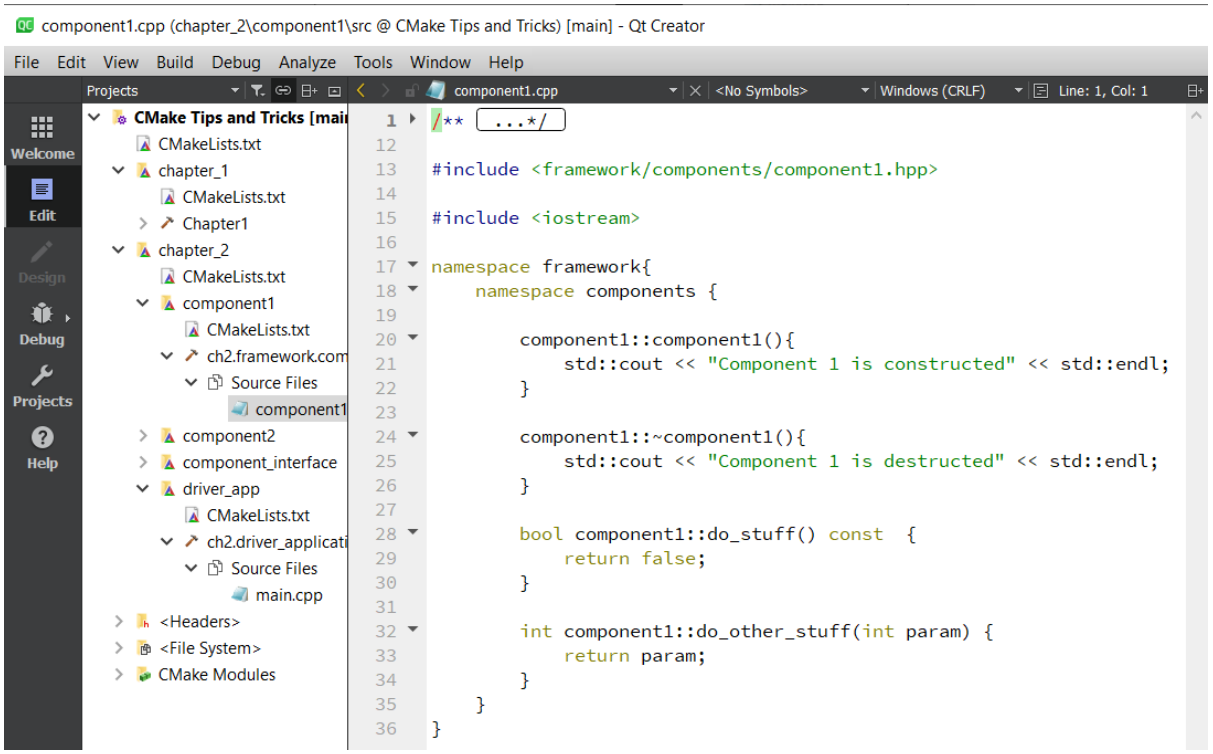

Options - Qt Creator
Filter
Kits
Environment
Text Editor
FakeVim
Help
C++
Qt Quick
Build & Run
Qbs
Debugger
Designer
Python
Analyzer
Version Control
Devices
Code Pasting
Kits
Kits
Qt Versions
Compilers
Debuggers
CMake
Name
Path
Auto-detected
CMake 3.21.1 (Qt) (Default)
D:\Qt_Creator\Tools\CMake_64\bi...
System CMake at C:\Program Files\CMake\bin\cmake.exe
C:\Program Files\CMake\bin\cma...
Manual
Add
Clone
Remove
Make Default
OK
Cancel
Apply

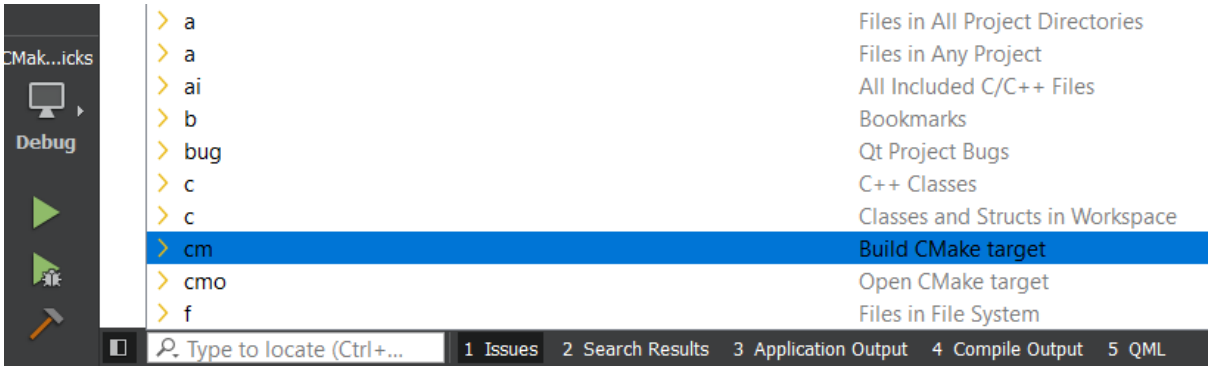

Manual
New CMake
Name:
New CMake
Path:
Browse...
Version:
Help file:
Browse...
Autorun CMake

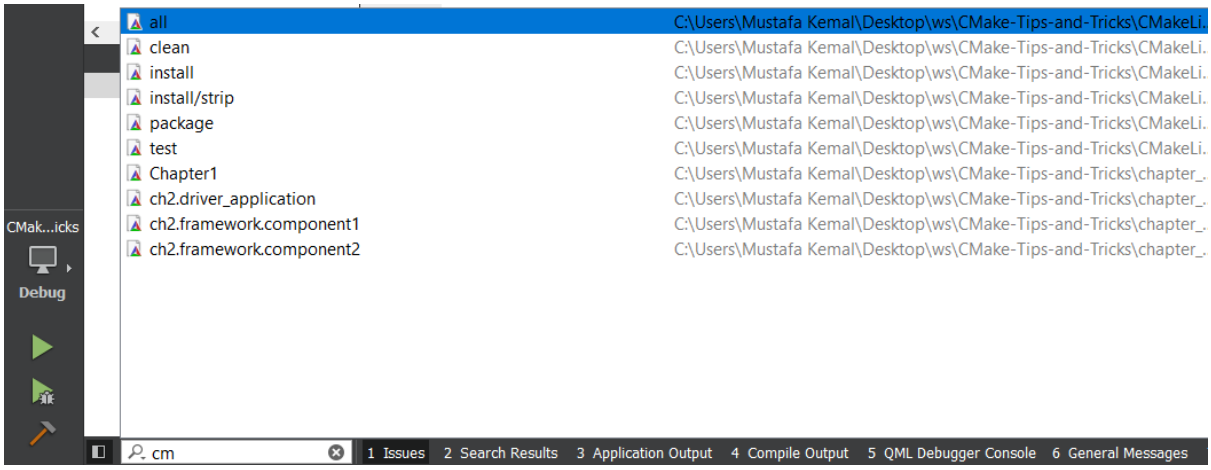

New File or Project - Qt Creator
Choose a template:
All Templates
Projects
Application (Qt)
Application (Qt Quick)
Application (Qt for Python)
Library
Other Project
Non-Qt Project
Import Project
Files and Classes
C/C++
Modeling
Qt
GLSL
General
Java
Python
Qt Widgets Application
Qt Console Application
Creates a Qt application for the desktop Includes a Qt Designer-based main window.
Preselects a desktop Qt for building the application if available.
Supported Platforms:
Desktop
Windows Runtime (Local)
Choose...
Cancel

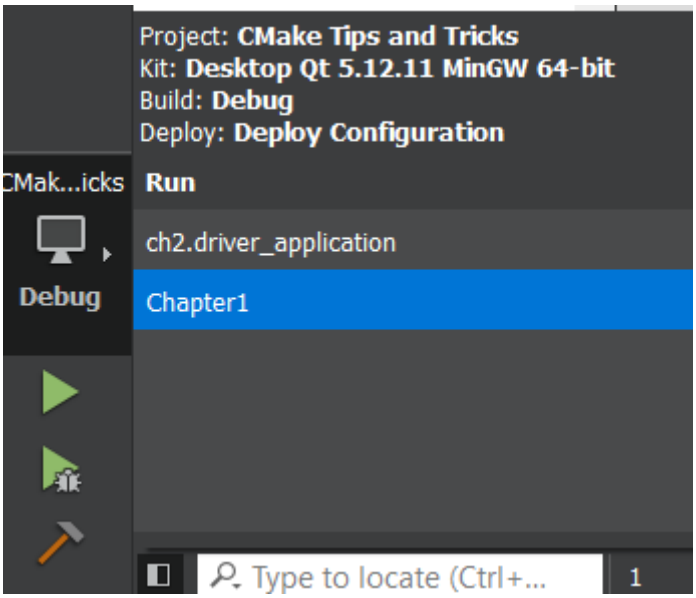

Qt Widgets Application
Define Build System
Location
Build System
Details
Translation
Kits
Summary
Build system:
CMake
Next
Cancel


main.cpp @ untitled1 - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
main.cpp
<Select Symbol>
Line: 12, Col: 1
untitled1
CMakeLists.txt
<File System>
CMakeLists.txt
main.cpp
mainwindow.cpp
mainwindow.h
mainwindow.ui
Welcome
Edit
Design
Debug
Projects
Help
Open Documents
main.cpp
#include "mainwindow.h"
#include <QApplication>
int main(int argc, char *argv[])
{
QApplication a(argc, argv);
MainWindow w;
w.show();
return a.exec();
}
Compile Output
Filter


component1.cpp (chapter_2\component1\src @ CMake Tips and Tricks) [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
component1.cpp
<No Symbols>
Windows (CRLF)
Line: 1, Col: 1
CMake Tips and Tricks [mai
CMakeLists.txt
chapter_1
CMakeLists.txt
Chapter1
chapter_2
CMakeLists.txt
component1
CMakeLists.txt
ch2.framework.com
Source Files
component1
component2
component_interface
driver_app
CMakeLists.txt
ch2.driver_applicati
Source Files
main.cpp
<Headers>
<File System>
CMake Modules
Welcome
Edit
Design
Debug
Projects
Help
/** ...*/
#include <framework/components/component1.hpp>
#include <iostream>
namespace framework{
namespace components {
component1::component1(){
std::cout << "Component 1 is constructed" << std::endl;
}
component1::~component1(){
std::cout << "Component 1 is destructed" << std::endl;
}
bool component1::do_stuff() const {
return false;
}
int component1::do_other_stuff(int param) {
return param;
}
}
}

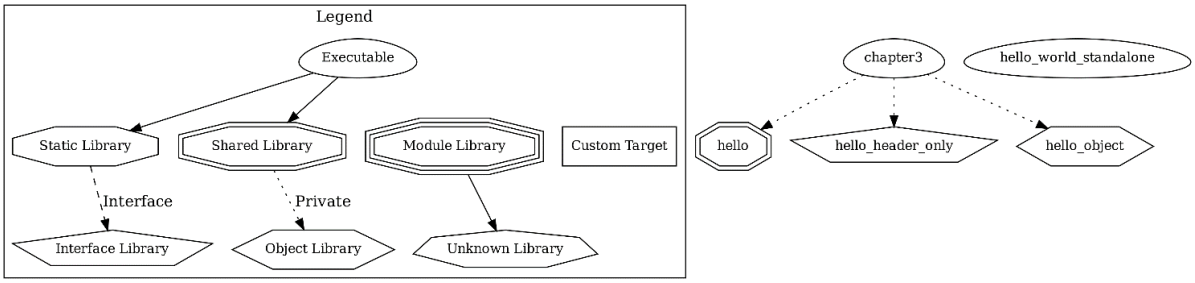

CMak...icks
Debug
a
Files in All Project Directories
a
Files in Any Project
ai
All Included C/C++ Files
b
Bookmarks
bug
Qt Project Bugs
c
C++ Classes
c
Classes and Structs in Workspace
cm
Build CMake target
cmo
Open CMake target
f
Files in File System
Type to locate (Ctrl+...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML


CMak...icks
Debug
all
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
clean
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
install
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
install/strip
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
package
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
test
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\CMakeLi.
Chapter1
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\chapter_.
ch2.driver_application
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\chapter_.
ch2.framework.component1
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\chapter_.
ch2.framework.component2
C:\Users\Mustafa Kemal\Desktop\ws\CMake-Tips-and-Tricks\chapter_.
cm
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
6 General Messages


Project: CMake Tips and Tricks
Kit: Desktop Qt 5.12.11 MinGW 64-bit
Build: Debug
Deploy: Deploy Configuration
CMak...icks
Run
Debug
ch2.driver_application
Chapter1
Type to locate (Ctrl+...
1

Chapter 3: Creating a CMake Project

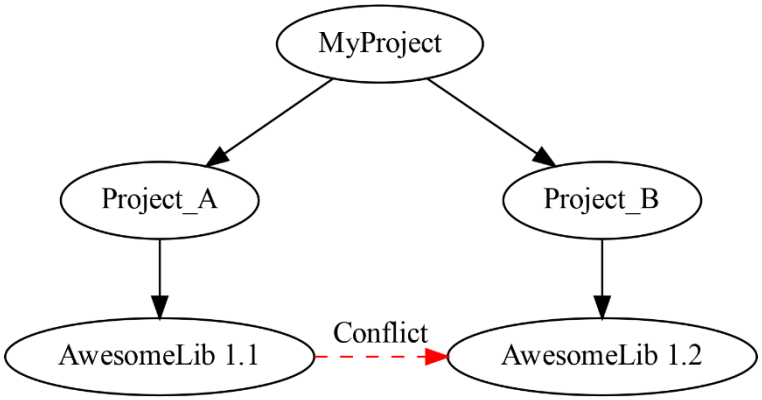
```
lrwxrwxrwx 1 conan    14 Sep 30 19:51 libhellod.so -> libhellod.so.1
lrwxrwxrwx 1 conan    18 Sep 30 19:51 libhellod.so.1 -> libhellod.so.1.0.0
-rwxr-xr-x 1 conan 91912 Sep 30 19:51 libhellod.so.1.0.0
```

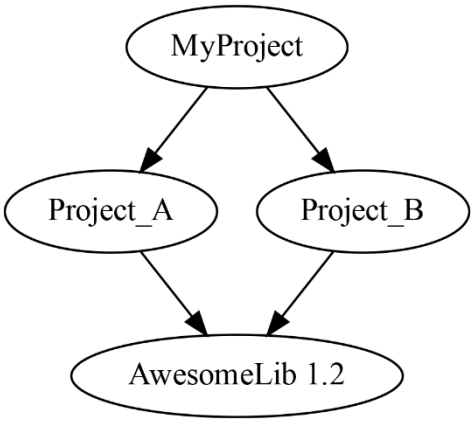


Chapter 4: Packaging, Deploying and Installing a CMake Project Effortlessly

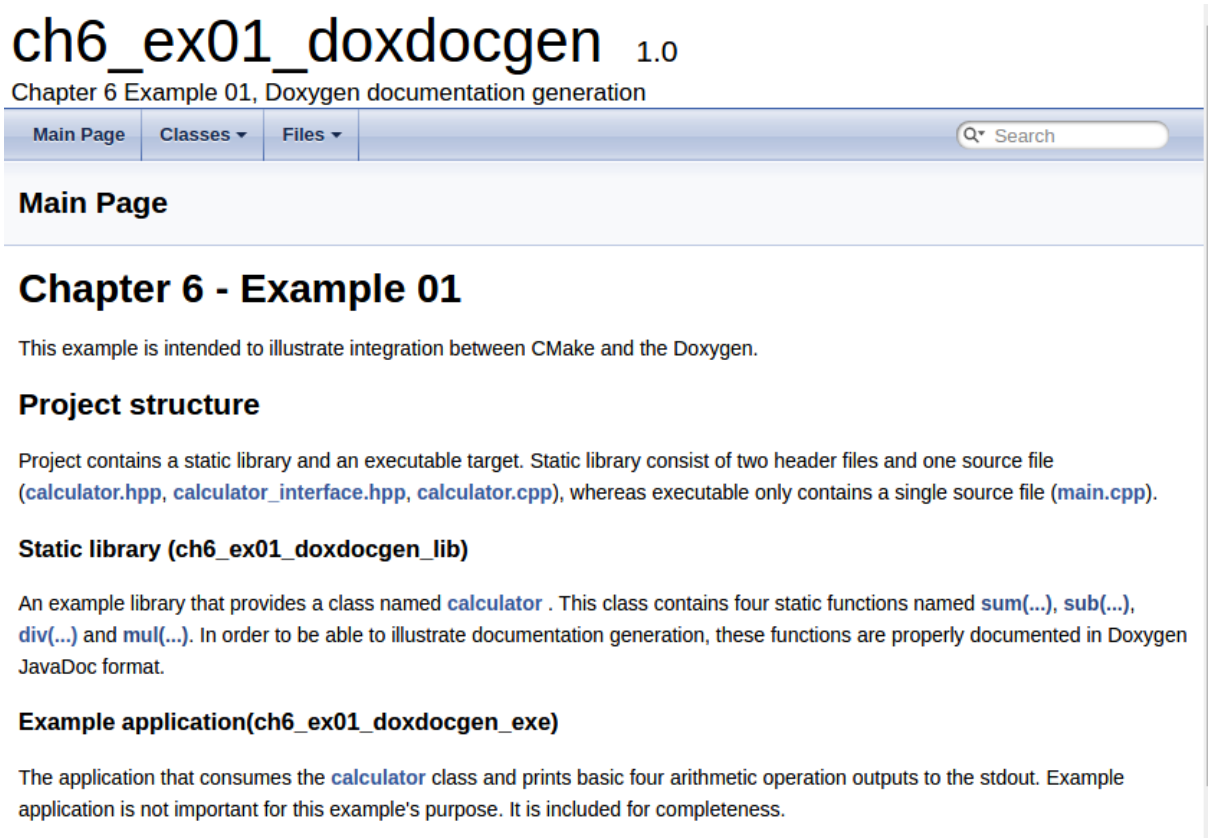
********** no images **********

Chapter 5: Integrating Third- Party Libraries and Dependency Management





Chapter 6: Automatically Generating Documentation with CMake



ch6_ex01_doxdocgen 1.0

Chapter 6 Example 01, Doxygen documentation generation

Main Page | Classes | Files | Search

Main Page

Chapter 6 - Example 01

This example is intended to illustrate integration between CMake and the Doxygen.

Project structure

Project contains a static library and an executable target. Static library consist of two header files and one source file (**calculator.hpp**, **calculator_interface.hpp**, **calculator.cpp**), whereas executable only contains a single source file (**main.cpp**).

Static library (ch6_ex01_doxdocgen_lib)

An example library that provides a class named **calculator** . This class contains four static functions named **sum(...)**, **sub(...)**, **div(...)** and **mul(...)**. In order to be able to illustrate documentation generation, these functions are properly documented in Doxygen JavaDoc format.

Example application(ch6_ex01_doxdocgen_exe)

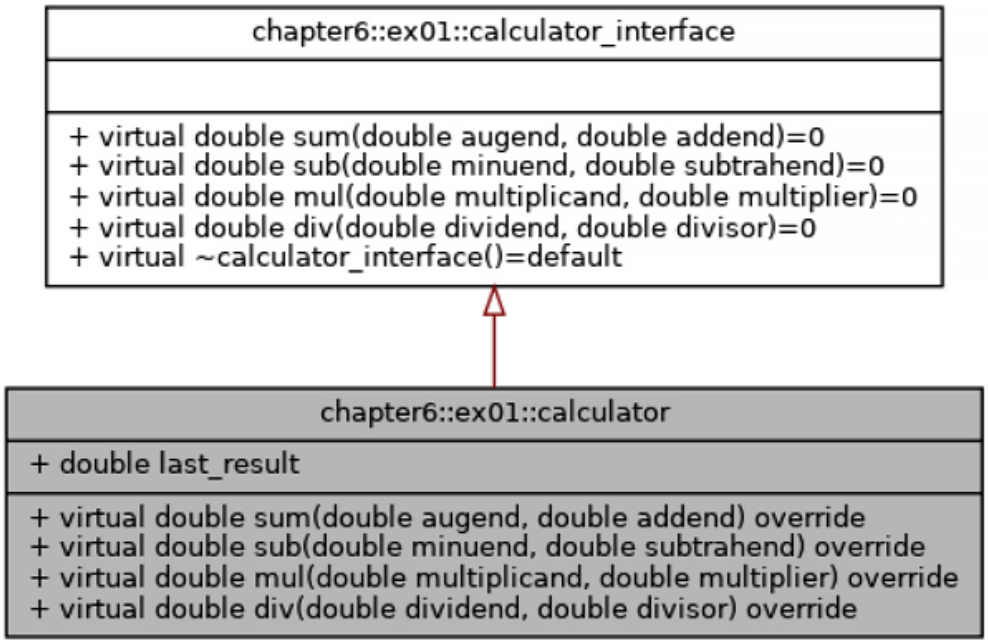
The application that consumes the **calculator** class and prints basic four arithmetic operation outputs to the stdout. Example application is not important for this example's purpose. It is included for completeness.

chapter6::ex01::calculator Class Reference

The basic 'calculator' class. More...

`#include <calculator.hpp>`

Inheritance diagram for chapter6::ex01::calculator:



Public Member Functions

virtual double	**sum** (double augend, double addend) override
virtual double	**sub** (double minuend, double subtrahend) override
virtual double	**mul** (double multiplicand, double multiplier) override
virtual double	**div** (double dividend, double divisor) override

Public Attributes

double	**last_result** {}

Member Function Documentation

◆ div()

double chapter6::ex01::calculator::div (double dividend,
double divisor
)

override virtual

Divide `dividend` with `divisor`

Parameters

[in] **dividend** The number to be divided by `divisor`

[in] **divisor** The number by which `divisor` is to be divided

Returns

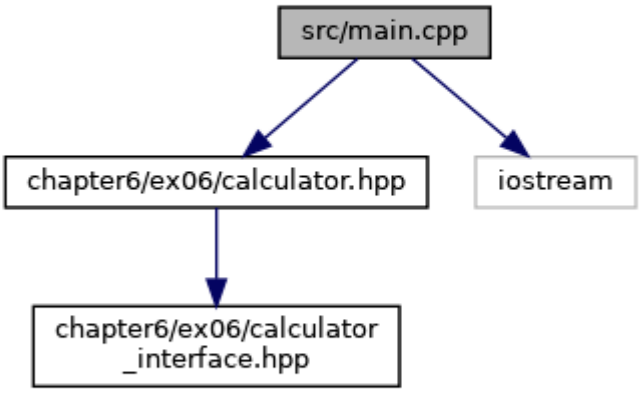
double Quotient of two numbers, `dividend` and `divisor`

Implements **chapter6::ex01::calculator_interface**.

main.cpp File Reference

```
#include <chapter6/ex06/calculator.hpp>
#include <iostream>
```

Include dependency graph for main.cpp:



Functions

int **main** (void)

Main entry point of the application. More...

◆ main()

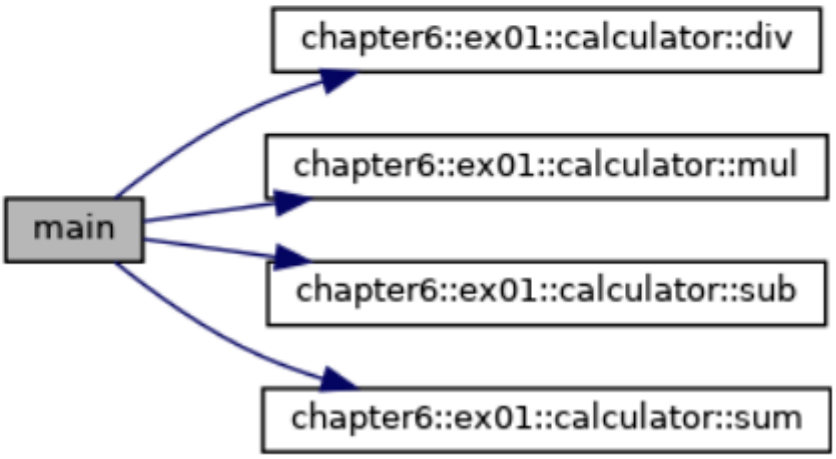
int main (void)

Main entry point of the application.

Returns

int Exit code

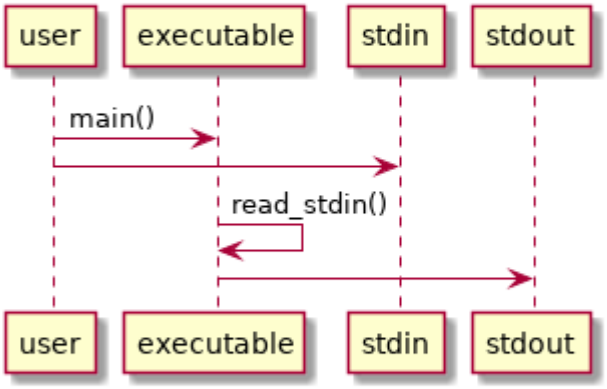
Here is the call graph for this function:



◆ main()

int main (void)

Main entry point of the application.



System Interaction Diagram

Returns

int Exit code

Chapter 7: Seamless Integration of Code Quality Tools with CMake

GCC Code Coverage Report

			Exec	Total	Coverage
Directory:	chapter_7/code_coverage_example/src/	**Lines:**	16	18	88.9 %
Date:	2021-11-20 21:03:35	**Branches:**	15	34	44.1 %
Legend:	low: < 75.0 % medium: >= 75.0 % high: >= 90.0 %				

File	Lines			Branches	
coverage_example/coverage_example.cpp		81.8 %	9 / 11	37.5 %	6 / 16
coverage_test.cpp		100.0 %	7 / 7	50.0 %	9 / 18

Generated by: GCOVR (Version 4.2)

LCOV - code coverage report

			Hit	Total	Coverage
Current view:	top level	**Lines:**	2410	8129	29.6 %
Test:	lcov.info	**Functions:**	3560	9301	38.3 %
Date:	2022-01-20 15:16:18				

Directory	Line Coverage		Functions	
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/build/_deps/catch2-src/single_include/catch2	30.5 %	1380 / 4532	35.3 %	463 / 1313
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/fixture_example/src	100.0 %	7 / 7	100.0 %	2 / 2
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/pass_fail_criteria/src	71.4 %	5 / 7	100.0 %	1 / 1
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/simple_test/src	100.0 %	2 / 2	100.0 %	1 / 1
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/test_discovery_example/src	100.0 %	6 / 6	100.0 %	5 / 5
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/timeout_example/src	100.0 %	3 / 3	100.0 %	1 / 1
/usr/include/x86_64-linux-gnu/c++/9/bits	50.0 %	2 / 4	50.0 %	1 / 2
9	50.0 %	73 / 146	37.3 %	124 / 332
9/bits	26.4 %	888 / 3363	36.7 %	2469 / 6727
9/ext	74.6 %	44 / 59	53.8 %	493 / 917

Generated by: LCOV version 1.14

LCOV - code coverage report

			Hit	Total	Coverage
Current view:	top level	**Lines:**	2410	8129	29.6 %
Test:	lcov.info	**Functions:**	3560	9301	38.3 %
Date:	2022-01-20 15:16:18				

Directory	Line Coverage		Functions	
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/build/_deps/catch2-src/single_include/catch2	30.5 %	1380 / 4532	35.3 %	463 / 1313
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/fixture_example/src	100.0 %	7 / 7	100.0 %	2 / 2
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/pass_fail_criteria/src	71.4 %	5 / 7	100.0 %	1 / 1
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/simple_test/src	100.0 %	2 / 2	100.0 %	1 / 1
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/test_discovery_example/src	100.0 %	6 / 6	100.0 %	5 / 5
/home/runner/work/CMake-Tips-and-Tricks/CMake-Tips-and-Tricks/chapter_7/timeout_example/src	100.0 %	3 / 3	100.0 %	1 / 1
/usr/include/x86_64-linux-gnu/c++/9/bits	50.0 %	2 / 4	50.0 %	1 / 2
9	50.0 %	73 / 146	37.3 %	124 / 332
9/bits	26.4 %	888 / 3363	36.7 %	2469 / 6727
9/ext	74.6 %	44 / 59	53.8 %	493 / 917

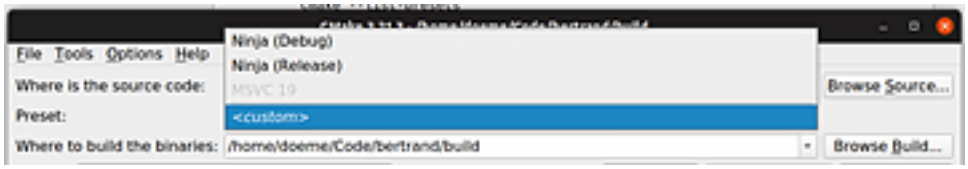
Generated by: LCOV version 1.14

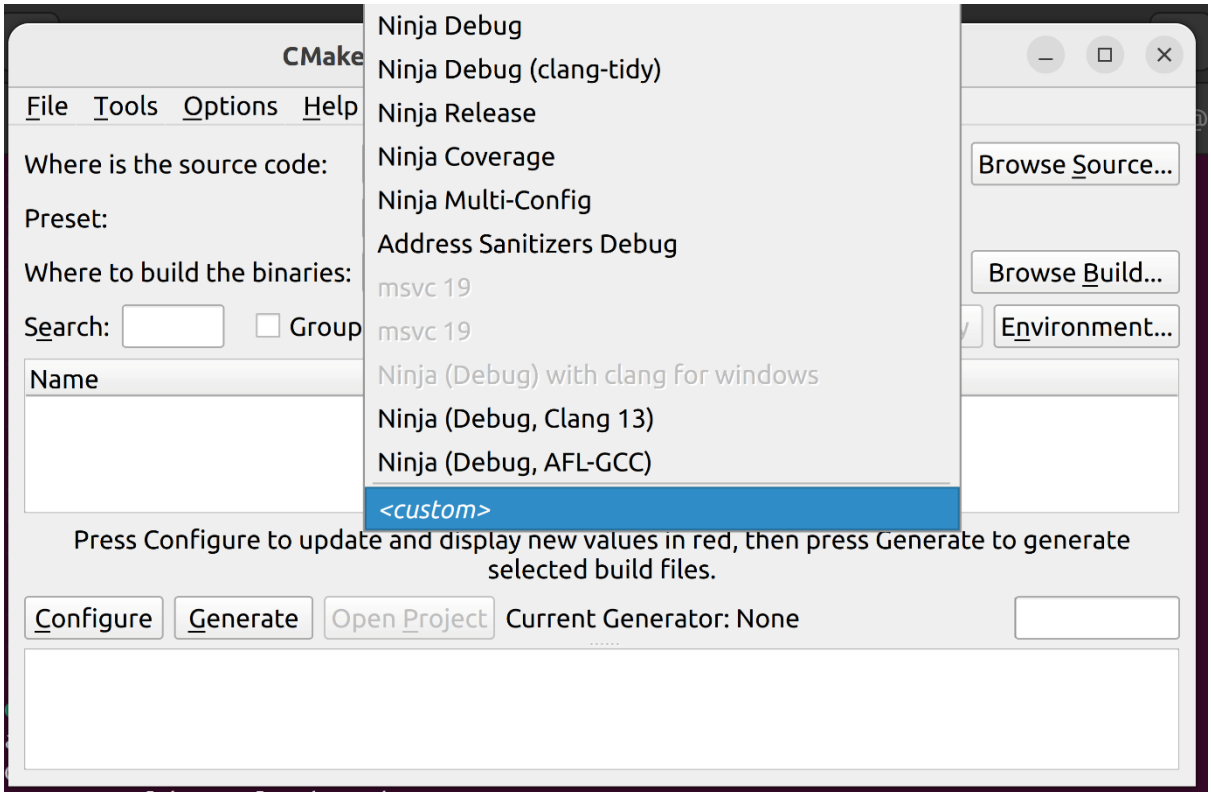
Coverage	Total lines	Items
	4876	ctest.exe
	4850	D:\at\CMake-Tips-and-Tricks\CMake-Tips-and-Tricks\build\chapter_7\test_discovery_example\Debug\ch7_test_discovery_example.exe
	9	D:\at\CMake-Tips-and-Tricks\CMake-Tips-and-Tricks\build\chapter_7\pass_fail_criteria\Debug\ch7_pass_fail_criteria.exe
	3	D:\at\CMake-Tips-and-Tricks\CMake-Tips-and-Tricks\build\chapter_7\fixture_example\Debug\ch7_fixture_sample.exe

Chapter 8: Executing Custom Tasks with CMake

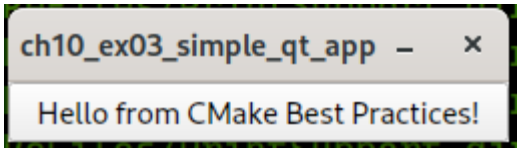
********** no images**********

Chapter 9: Creating Reproducible Build Environments

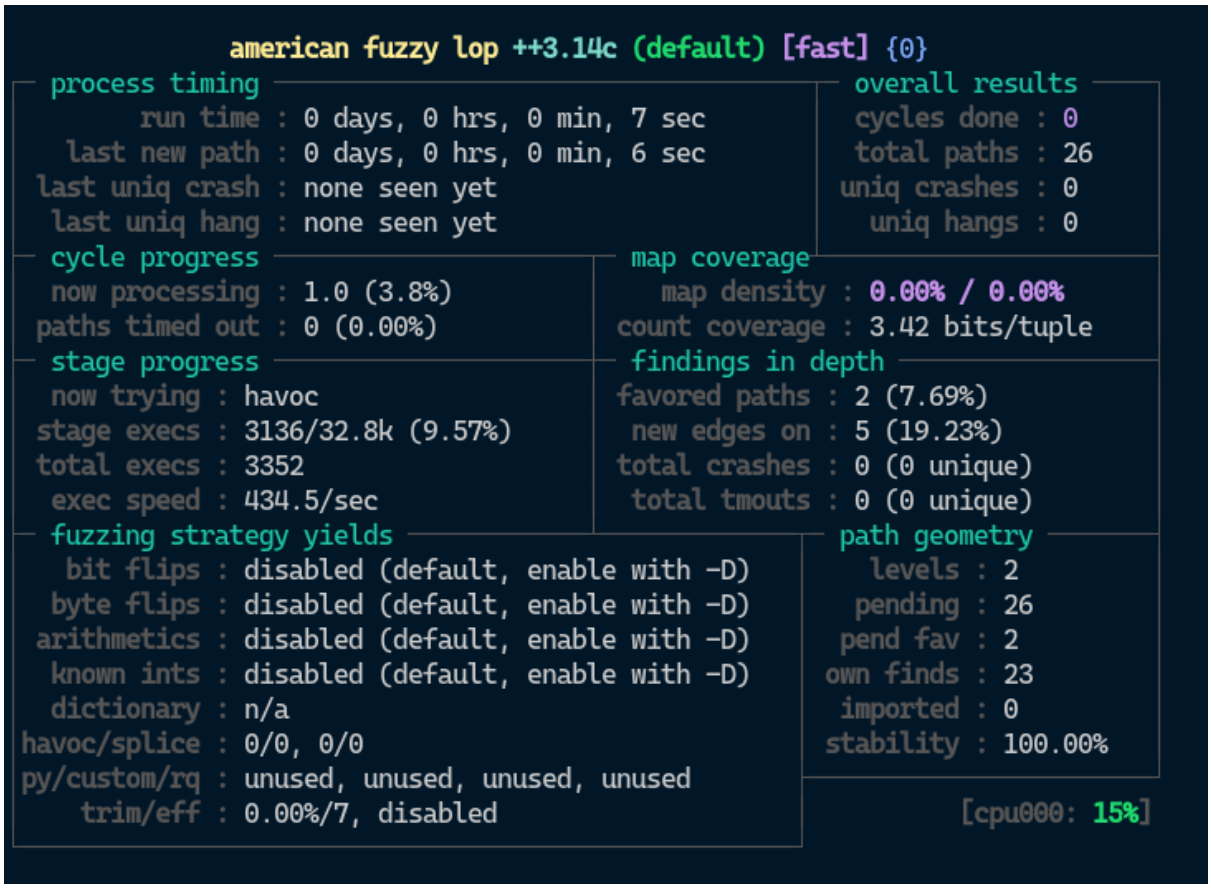


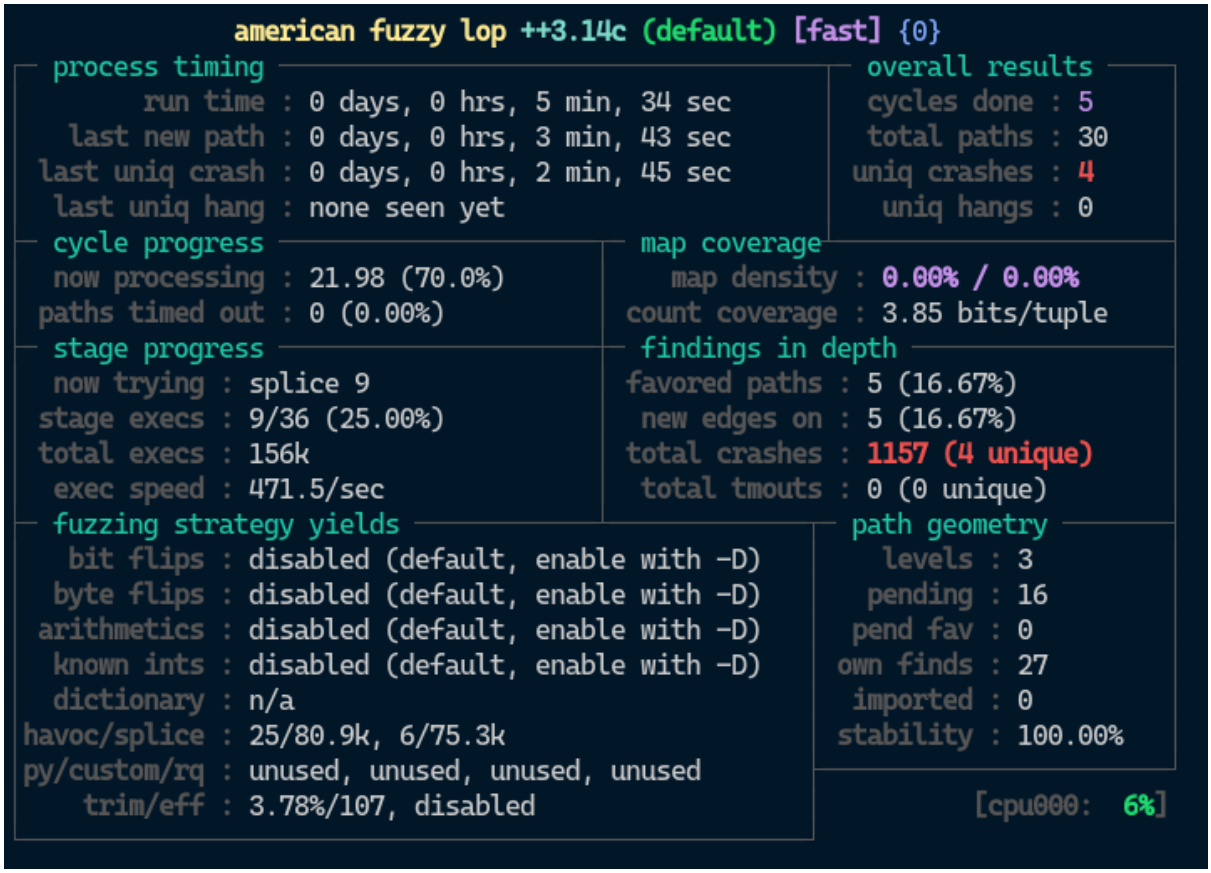


Chapter 10: Handling Distributed Repositories and Dependencies in a Super-build



Chapter 11: Automated Fuzzing with CMake






```
18:03 $ ls
README.txt
id:000000,sig:11,src:000021+000026,time:90095,op:splice,rep:4
id:000001,sig:11,src:000021+000026,time:92289,op:splice,rep:4
id:000002,sig:11,src:000021+000026,time:92325,op:splice,rep:16
id:000003,sig:11,src:000001+000028,time:168241,op:splice,rep:2
```

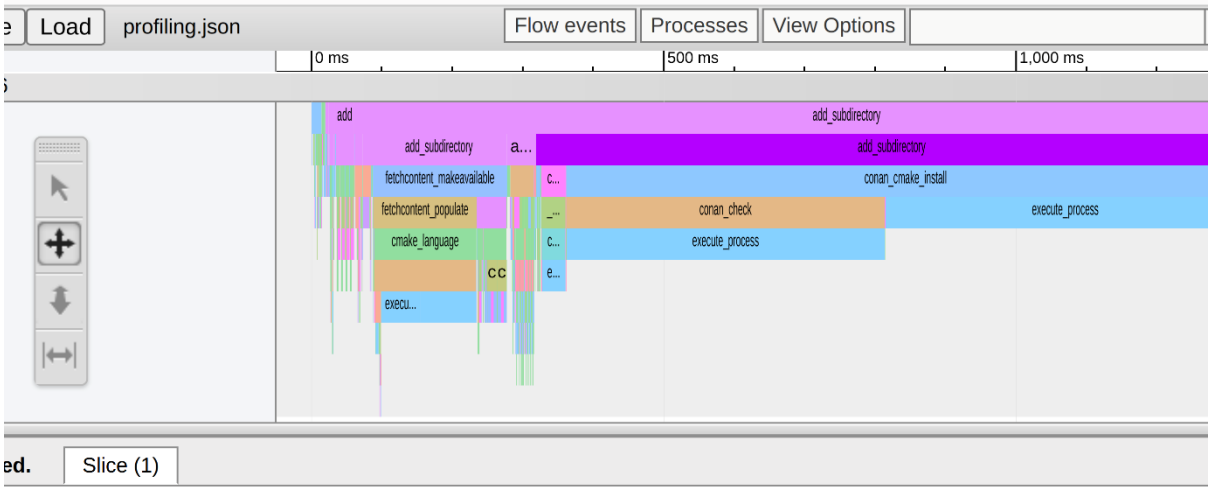
Chapter 12: Cross-Platform-Compiling Custom Toolchains

********** no images**********

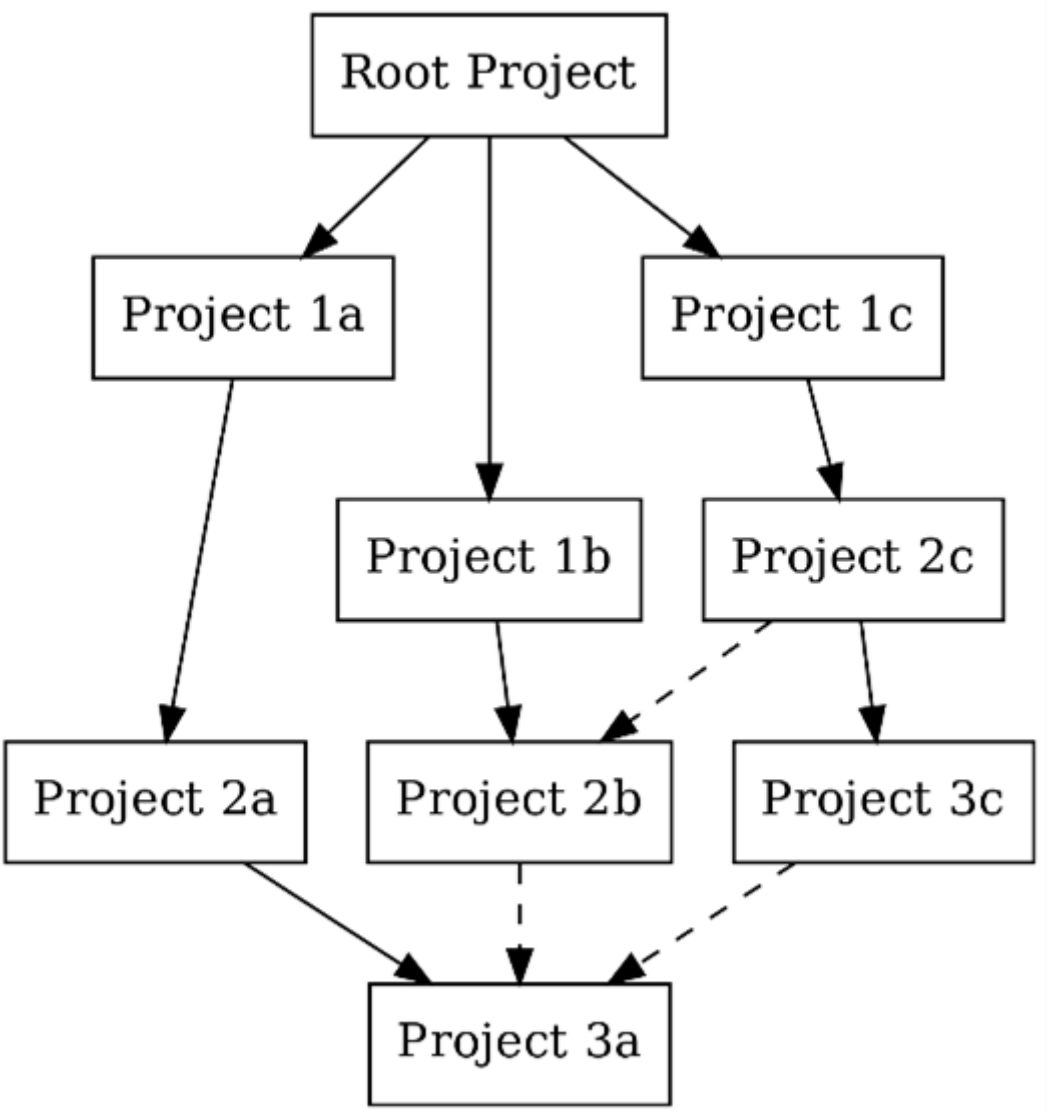
Chapter 13: Reusing CMake Code

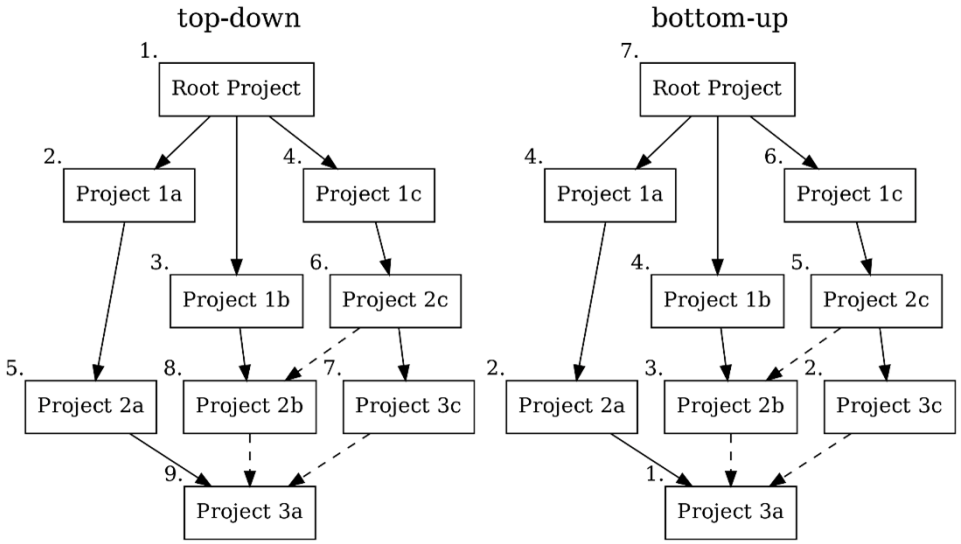
********** no images**********

Chapter 14: Optimizing and Maintaining CMake Projects



Chapter 15: Migrating to CMake




top-down
1.
Root Project
2.
Project 1a
4.
Project 1c
3.
Project 1b
6.
Project 2c
5.
Project 2a
8.
Project 2b
7.
Project 3c
9.
Project 3a
bottom-up
7.
Root Project
4.
Project 1a
6.
Project 1c
4.
Project 1b
5.
Project 2c
2.
Project 2a
3.
Project 2b
2.
Project 3c
1.
Project 3a

Chapter 16: Contributing to CMake and Further Reading Material

********** no images**********