|  |
| --- |
|  |
|  |
| conan 삭제 |
|  |
| conan install |
|  |
| CMake+Conan 설명 |
| https://www.youtube.com/watch?v=cwg886syRcU |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| 이 conanfile.py를 편집한다. |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| ... |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| https://github.com/CppIndia-UserGroup/CppIndia-SessionDocuments/tree/master/19-12-2020 |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| 샘플 |
| https://github.com/conan-io/training |
| * conanfile.py is a Conan "recipe". It declares dependencies, how to build a package from sources. The same recipe can be used to manage different configurations, like different platforms, compilers, build types, etc * conanfile.txt is a text simplification of conanfile.py, that can be used exclusively to consume dependencies, but not to create packages. conanfile.py can be used both to consume dependencies and to create packages * a profile file is a text file containing configuration values like os=Windows and compiler=gcc. You can pass these values in the command line, but it is better to have them in files, easier to manage and more convenient. * settings.yml is a declaration of what values can the settings take. To validate the inputs and make sure there are no typos, and use a common set of configurations so people can collaborate together.   I suggest following the tutorials in the docs, like <https://docs.conan.io/en/latest/getting_started.html>, or if you are into video-format, this free training is good: <https://academy.jfrog.com/path/conan>  Regarding the versions, you need to use those defined in the settings, for Visual Studio you need to use 14, 15, etc. The new msvc compiler setting, experimental, will be using the compiler version, like 19.xx. In general, it is not needed to specify the compiler version down to the patch, because this is mostly for the binary model, and it is typically not needed to go to that level. If you want to learn how to customize the settings value, [read this section](https://docs.conan.io/en/latest/extending/custom_settings.html) |
|  |
|  |
| install packages from the new conan center remote |
| conan install spdlog/1.11.0 |
| conan install boost/1.72.0 |
|  |
|  |
|  |
| CMake 강좌 |
| https://www.youtube.com/watch?v=QhKkZ\_7Wgfg |
|  |
|  |
| Visual Studio 2022에서는 bullet CMake가 된다. |
|  |
|  |
| 2022에서 이것이 된다. |
| D:\git\projects\modern-cmake\examples |
|  |
| Conan-project 테스트 |
| D:\git\projects\examples2\tutorial\consuming\_packages\simple\_cmake\_project |
|  |
| https://docs.conan.io/en/2.0/tutorial/consuming\_packages/build\_simple\_cmake\_project.html |
| “C:\Program\_Files\Microsoft\_Visual\_Studio\2022\Professional\Common7\IDE\CommonExtensions\Microsoft\CMake\CMake\share\cmake-3.24\Modules |
| FindPackageHandleStandardArgs.cmake:230(message) |
| Could Not find ZLIB(missing ZLIB\_LIBRARY ZLIB\_INCLUDE\_DIR) |
|  |
|  |
|  |
|  |
|  |
|  |