

CMAKE Cheatsheet

The Basics

Basic commands

Project definition	<code>project (<NAME>)</code>
cmake version requirement	<code>cmake_minimum_required(VERSION 2.8.7)</code>
Add a subdirectory to cmake	<code>add_Subdirectory(src)</code>

Executables and Libraries

Targets

Add library target	<code>Add_Library (data \${DATA_SOURCES})</code>
Add dependency libraries (for library or an executable)	<code>Target_Link_Libraries (test_generator util data fix fixml)</code>
Add executable target	<code>add_executable(\${CPP_EXE} \${CPP_FILE})</code>
Add an executable target for each cpp file	<code>foreach (CPP \${SAMPLES_SRCS}) get_filename_component(EXE \${CPP} NAME_WE) add_executable (\${EXE} \${FILE}) endforeach(CPP_FILE)</code>

Files

Filenames

Get files by their names using wildcard (here all cpp files)	<code>File (GLOB <VAR_NAME> *.cpp)</code>
Get filename without extension	<code>get_filename_component(FNAME \${CPP} NAME_WE)</code>
Get name without directory	<code>get_filename_component(FNAME \${CPP} NAME)</code>
Get file extension (longest)	<code>get_filename_component(FEXT \${CPP} EXT)</code>
Get file absolute path	<code>get_filename_component(FPATH \${CPP} ABSOLUTE)</code>
Get file absolute path with symlinks resolved	<code>get_filename_component(FPATH \${CPP} REALPATH)</code>
Get file directory	<code>get_filename_component(FDIR \${CPP} DIRECTORY)</code>

C++

Compiler

Define C++ standard requirement of the compiler (here c++14)	<code>set(CMAKE_CXX_STANDARD 14) set(CMAKE_CXX_STANDARD_REQUIRED ON)</code>
Add compilation flag	<code>add_definitions ("-std=c++11")</code>
Add include directory	<code>Include_Directories(\${Xerces_INCLUDE_DIRS})</code>

Tests

Tests

TODO	Command
------	---------

Packaging and Installation

Packaging

TODO	Command
------	---------

Installation

TODO	Command
------	---------