## cmake-commands

### 1.1 [add\_custom\_command](https://cmake.org/cmake/help/v3.2/command/add_custom_command.html)

(1) **command location:** build\_pre.cmk.

(2) **function:** adds a custom command to a target, like shell, python command.

(3) **commonly used form:**

add\_custom\_command(TARGET target

PRE\_BUILD | PRE\_LINK | POST\_BUILD

COMMAND command1 [ARGS] [args1...]

[COMMAND command2 [ARGS] [args2...] ...]

[WORKING\_DIRECTORY dir])

1. A **target** can be exe, lib or generated by [add\_custom\_target()](https://cmake.org/cmake/help/v3.2/command/add_custom_target.html).
2. **PRE\_BUILD:** Run before any other rules are executed within the target. This is supported only on Visual Studio 7 or later. For all other generators PRE\_BUILD will be treated as PRE\_LINK. For instance, when build **libmsg.a**, the COMMAND will execute after object files generated and before linking it. For macro.h will not generated before compiling object files, error will occur.

**PRE\_LINK:** Run after sources have been compiled but before linking the binary or running the librarian or archiver tool of a static library.

**POST\_BUILD:** Run after the target built.

(c) **WORKING\_DIRECTORY:** Appoint where the command execute. If the command is touch test.txt and dir is tmp/, then test.txt will generate in tmp/.

1. **note:**
2. This command should be used after target generated command. For example:

add\_executable(demo demo.c)

add\_custom\_command(TARGET target

PRE\_BUILD

COMMAND echo “test”)

(b) This command run when make the generated Makefile.