

# Assert using Boost library

---

## Methods:

---

- BOOST\_ASSERT :
  - gives
    - nothing if the expression is *true*.
    - the error (without message) if the expression is *false*.
  - syntax: BOOST\_ASSERT(expr)
  - example ([/F:/Coding/github\\_repos/cpp-playground/libs/boost/examples/assert/assert.cpp](/F:/Coding/github_repos/cpp-playground/libs/boost/examples/assert/assert.cpp))
- BOOST\_ASSERT\_MSG :
  - gives:
    - nothing if the expression is *true*.
    - the error (with custom message) if the expression is *false*.
  - syntax: BOOST\_ASSERT\_MSG(expr, msg)
  - example ([/F:/Coding/github\\_repos/cpp-playground/libs/boost/examples/assert/assert\\_msg.cpp](/F:/Coding/github_repos/cpp-playground/libs/boost/examples/assert/assert_msg.cpp))
- BOOST\_VERIFY :
  - gives:
    - nothing if the expression is evaluated correctly.
    - the error (without message) if the expression is NOT VALID.
  - syntax: BOOST\_VERIFY(expr)
  - example ([/F:/Coding/github\\_repos/cpp-playground/libs/boost/examples/assert/verify.cpp](/F:/Coding/github_repos/cpp-playground/libs/boost/examples/assert/verify.cpp))
- BOOST\_VERIFY\_MSG :
  - gives:
    - nothing if the expression is evaluated correctly.
    - the error (with custom message) if the expression is NOT VALID.
  - syntax: BOOST\_VERIFY\_MSG(expr, msg)
  - example ([/F:/Coding/github\\_repos/cpp-playground/libs/boost/examples/assert/verify\\_msg.cpp](/F:/Coding/github_repos/cpp-playground/libs/boost/examples/assert/verify_msg.cpp))

- Using macros: To disable, mention before `#include <boost/assert.hpp>` line in the source/header file. example ([/F:/Coding/github\\_repos/cpp-playground/libs/boost/examples/assert/assert\\_enable\\_disable.cpp](/F:/Coding/github_repos/cpp-playground/libs/boost/examples/assert/assert_enable_disable.cpp))

- `#define BOOST_DISABLE_ASSERTS` : disable assert methods
- `#define NDEBUG` : disable assert methods
- `#define BOOST_ENABLE_ASSERTS` : enable assert methods. Enabled by default.

It affects both the `assert` & `verify` (with/without msg)

- `assert_custom_func`
  - gives:
    - nothing if the expression is *true*.
    - the error (with custom message) if the expression is *false*.
  - doesn't use any library
  - arg-1: bool expression
  - arg-2: lambda func