## Contributing

This library is free, and will stay free but needs your support to sustain its development. There are lots of [\*\*new features\*\*](doc/markdown/roadmap.md) and maintenance to do. If you work for a company using \*\*doctest\*\* or have the means to do so, please consider financial support.

[![Patreon](https://cloud.githubusercontent.com/assets/8225057/5990484/70413560-a9ab-11e4-8942-1a63607c0b00.png)](http://www.patreon.com/onqtam)

[![PayPal](https://www.paypalobjects.com/en\_US/i/btn/btn\_donate\_LG.gif)](https://www.paypal.me/onqtam/10)

## Pull requests

Consider opening an issue for a discussion before making a pull request to make sure the contribution goes smoothly.

All pull requests should be made against the ```dev``` branch because the ```master``` is the stable one with the latest release.

If you're going to change something in the library itself - make sure you don't modify ```doctest/doctest.h``` because it's generated from ```doctest/parts/doctest\_fwd.h``` and ```doctest/parts/doctest.cpp``` - they get concatenated by CMake - so make sure you do a CMake build after you modify them so the ```assemble\_single\_header``` target gets built. Also take into consideration how the change affects the code coverage - based on the project in ```examples/all\_features```. Also update any relevant examples in the ```examples``` folder.

This framework has some design goals which must be kept. Make sure you have read the [\*\*features and design goals\*\*](doc/markdown/features.md) page.

If your changes also change the output of the library - you should also update the reference output for the tests or otherwise the CI builds (```travis``` and ```appveyor```) will fail when they compare the latest output to the outdated reference output (which is committed in the repository). To do this run CMake with the ```DOCTEST\_TEST\_MODE``` variable set to ```COLLECT``` (making the new reference output) and then run ```ctest``` and commit the changed (or newly created) ```.txt``` files in the ```test\_output``` folders too. The default ```DOCTEST\_TEST\_MODE``` is ```COMPARE```.

Example: ```cmake -DDOCTEST\_TEST\_MODE=COLLECT path/to/sources && cmake --build . && ctest```

Code should be formatted with a recent-enough ```clang-format``` using the config file in the root of the repo (or I will do it...)

Testing with compilers different from GCC/Clang/MSVC (and more platforms) is something the project would benefit from.

---------------

[Home](readme.md#reference)

<p align="center"><img src="scripts/data/logo/icon\_2.svg"></p>