

## Exercise sheet 1

2022-10-20

Due date: 2022-10-27 16:59

The goal of this exercise sheet is to get you used to the workflow of C++ projects with git, compilers and source files. Refer to your repository's README.md file for more detailed instructions on setting everything up properly.

### Exercise 0:

Gain access to the course group on GitLab and install the required tools

- submit your GitLab username on Moodle so we can grade you
- request to join the GitLab group:  
<https://gitlab.lrz.de/cppcourse/ws2022/waiting-room>
- make sure you have installed all the necessary tools (git, compiler, ...)
- set up your preferred development environment (IDE or editor)

### Exercise 1:

Set up the assignments repository

- clone the repository to your machine

```
$ git clone git@gitlab.lrz.de:cppcourse/ws2022/#your-project-name#.git
```

- configure pulling from upstream (to get our updates for future assignments)
- verify your setup

```
$ git remote -v
```

### Exercise 2:

Build the provided project manually - and fix the missing parts. Invoke the compiler directly in a shell, like you learned in the exercise session.

- in hw01/ you will see some source files
- build a shared library from `library.cpp`:  

```
g++ -std=c++20 -Wall -Wextra -shared -o liblibrary.so library.cpp
```
- create a header file for this library and include it in `hw01.cpp`, so the library function can be accessed

- build `hw01.cpp` and link it to the library:

```
g++ -std=c++20 -Wall -Wextra -c -o hw01.o hw01.cpp
```

```
g++ -std=c++20 -Wall -Wextra -L . hw01.o -llibrary -Wl,-rpath . -o hw01
```

- now running `./hw01` should print a number, success!

## Exercise 3:

We use `doctest` to validate the solutions of upcoming homeworks.

Install `doctest` by manually cloning it.

Doctest can be cloned from GitHub: <https://github.com/doctest/doctest/>.

When one defines `DOCTEST_CONFIG_IMPLEMENT_WITH_MAIN`, doctest will define its own `main` function.

When a program contains a `main` function, it can be executed directly.

- Build `test.cpp` while including `doctest` and link it to library:

```
g++ -std=c++20 -Wall -Wextra -Wl,-rpath .  
-I include-path-to-doctest -o hw01test test.cpp -L . -llibrary
```

- when you execute `./hw01test`, `doctest` will print out pretty results
- fix the code until the test is happy!