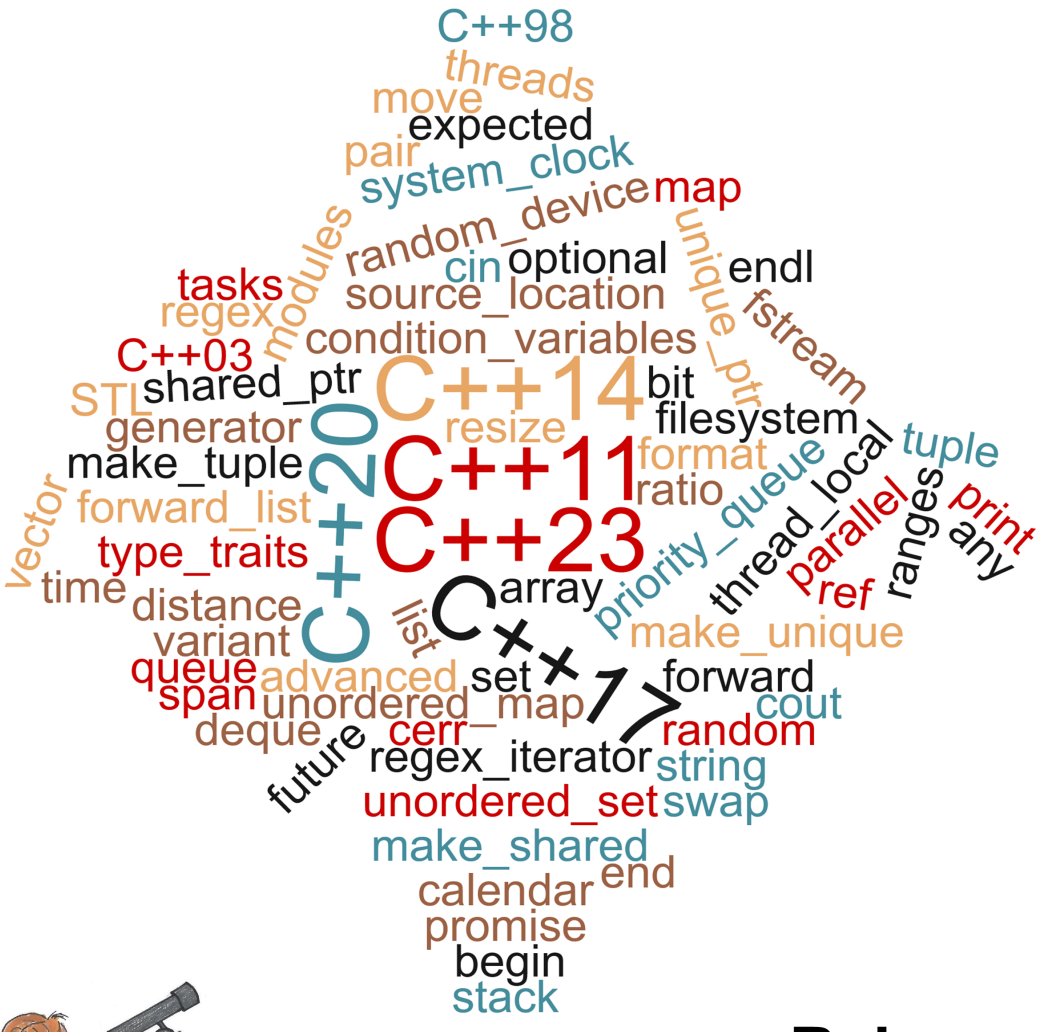
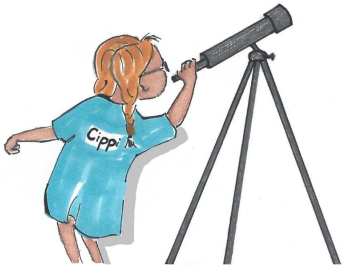
The C++ Standard Library

Fourth Edition includes C++23





Rainer Grimm

ModernesCpp.com

The C++ Standard Library

What every professional C++ programmer should know about the C++ standard library.

Rainer Grimm

This book is for sale at http://leanpub.com/cpplibrary

This version was published on 2023-03-11

This is a Leanpub book. Leanpub empowers authors and publishers with the Lean Publishing process. Lean Publishing is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

Contents

1. Introduction

Purpose of this Book

The C++ Standard Library is a quick reference to the standard library of the current C++23 standard ISO/IEC 14882:2023[1]. C++23 has more than 2100 pages and follows the big C++20 standard. In contrast, C++23 and C++17 are neither a big nor small C++ standards. C++14 is a small addition to C++11. C++11 had more than 1,300 pages and was published in 2011. That was 13 years after the first and only C++ standard, C++98. Of course, there is also C++03, published in 2003. But C++03 is considered a bug-fix release.

This quick reference aims is to provide a concise reference to the C++ standard library. This book assumes that you are familiar with C++. If so, you will get the most benefit out of this book. If C++ is new to you, you should start with a textbook about core C++. Once you have mastered a book about the core language, you can make your next big step by reading this book. To make your job easier, I have provided many short code snippets to connect theory and practice.

Index

The book should be a reference for C++ and should, therefore, have an index. Leanpub does not support the creation of an index. So I've made it based on regular expressions, naming conventions, a lot of python magic, and a long table that I had to split for each page. Here is the problem. The index is only fully available in the pdf format of the book.

Conventions

I promise only a few conventions.

Special Fonts

Italic
: I use *Italic* if something is essential.

`Monospace`
: I use `Monospace` for code, instructions, keywords, and names of types, variables, functions, and classes.

[1] https://www.iso.org/standards.html

Special Boxes

I use boxes for unique information, tips, and warning.

Information headline

Information text.

Tip headline

Tip description.

Warning headline

Warning description.

Source Examples

I don't particularly appreciate using directives and declarations because they hide the library's namespace. I use them so that the origin can always be deduced from the using *directive* (`using namespace std;`) or the *using* declaration (`using std::cout;`). Still, because of the limited length of a page, I have to use them from time to time.

Only header files of the featured functionality are shown in the code snippets. `true` or `false` is displayed in the output code snippets for boolean values, and `std::boolalpha` is not used. When your compiler supports the modularized standard library in C++23, you can replace the headers with an `import std` statement.

Source Code

To be concise, I only present short code snippets in this book. The name of the entire program is in the first line of the code snippet.

Value versus Object

I call instances of fundamental data types *values*, which C++ inherited from C. Instances of more advanced types, which often consist of fundamental types, are called *objects*. Objects are typically instances of *user-defined* types or containers.

Acknowledgments

First, I want to thank Alexandra Follenius, the lector at O'Reilly, for the German book C++ Standardbibliothek[2]. The German book is the ancestor of this book. For my book *C++ Standardbibliothek* Karsten Ahnert, Guntram Berti, Dmitry Ganyushin, Sven Johannsen, Torsten Robitzki, Bart Vandewoestyne, and Felix Winter were very valuable proofreaders. A lot of thanks to all of them.

I started a request in my English blog for translating this book to English www.ModernesCpp.com[3]. I received a much higher response than I expected. Special thanks to all of you, including my son Marius, the first proofreader.

Here are the alphabetically ordered names: Mahesh Attarde, Rick Audet, Pete Barrow, Michael Ben-David, Dave Burns, Alvaro Fernandez, Juliette Grimm, George Haake, Clare Macrae, Arne Mertz, Ian Reeve, Jason Turner, Bart Vandewoestyne, Ivan Vergiliev, and Andrzej Warzynski.

Further Information

The idea of the book is relatively easy to paraphrase: "What every professional C++ programmer should know about the C++ standard library." Because of this intention, I left many answers unanswered; therefore, I provide you with the links to the details at the beginning of each new topic. The link will refer to the excellent online resource www.cppreference.com[4].

Cippi

Let me introduce Cippi. Cippi will accompany you in this book. I hope you like her.

[2] http://shop.oreilly.com/product/9783955619688.do
[3] http://www.modernescpp.com/index.php/do-you-wan-t-to-proofread-a-book
[4] http://en.cppreference.com/w/

I'm Cippi: curious, clever and - yes - feminine!

About Me

I've worked as a software architect, team lead, and instructor since 1999. In 2002, I created company-intern meetings for further education. I have given training courses since 2002. My first tutorials were about proprietary management software, but I began teaching Python and C++ soon after. I like to write articles about C++, Python, and Haskell in my spare time. I also like to speak at conferences. I publish weekly on my English blog Modernes Cpp[5], and the German blog[6], hosted by Heise Developer.

Since 2016, I have been an independent instructor giving seminars about modern C++ and Python. I have published several books in various languages about modern C++ and, in particular, about concurrency. Due to my profession, I always search for the best way to teach modern C++.

[5] https://www.modernescpp.com/
[6] https://www.grimm-jaud.de/index.php/blog

Rainer Grimm