mdBook Markdown books

START HERE

`mdbook init <dir>`
Scaffold a new book

`book.toml`
Book configuration

`mdbook serve --open`
Live preview in browser

`mdbook watch`
Rebuild on change

`mdbook build`
Render to book/

`mdbook test`
Check Rust snippets

`mdbook clean`
Delete the output

`src/SUMMARY.md`
Table of contents

src/*.md chapters → `SUMMARY.md` → **Preprocessors** links, index → `mdbook build` → **Renderers** html, custom → `writes` → **book/** static site

CONCEPTS

SUMMARY.md decides which chapters exist and their order; a Markdown file not listed there is not part of the book.

Install with cargo install mdbook, or grab a prebuilt binary from the GitHub releases.

Third-party features come as mdbook-<name> executables, enabled via [preprocessor.<name>] or [output.<name>] in book.toml.

CREATE AND BUILD

`mdbook init`
Create a book in the current directory

`mdbook init <dir> --title "<title>"`
Create a book in a new directory

`mdbook init --force`
Skip the title and .gitignore prompts

`mdbook init --ignore git`
Also write a .gitignore for book/

`mdbook init --theme`
Copy the default theme to edit it

CREATE AND BUILD (cont.)

`mdbook build`
Render the book to book/

`mdbook build <dir>`
Build a book in another directory

`mdbook build -o`
Build, then open in the browser

`mdbook build -d <dir>`
Write output to a different directory

`mdbook clean`
Delete the rendered output

`mdbook clean -d <dir>`
Delete a specific output directory

PREVIEW

`mdbook serve`
Serve and rebuild on every change

`mdbook serve -o`
Serve and open the browser

`mdbook serve -p <port>`
Use another port (default 3000)

`mdbook serve -n <host>`
Bind to another hostname
Use 0.0.0.0 to reach the preview from other machines on the network.

`mdbook serve --watcher native`
Use OS file events instead of polling

PREVIEW (cont.)

`mdbook watch`
Rebuild on change without a server

Only the .gitignore in the book root excludes files from triggering rebuilds.

TEST AND TOOLING

`mdbook test`
Compile and run all Rust code blocks

`mdbook test -c <chapter>`
Test one chapter by name or path

`mdbook test -L <deps-dir>`
Add a library search path for rustdoc
Point it at target/debug/deps to test snippets that use your crate.

`mdbook completions <shell>`
Print a shell completion script

`mdbook help`
List commands

`mdbook --version`
Show the installed version

SUMMARY.MD

`# Summary`
Optional title (ignored by the parser)

`[Intro](intro.md)`
Prefix chapter, unnumbered, before the list

SUMMARY.MD (cont.)

`- [Chapter](chapter.md)`
Numbered chapter

`    - [Sub](sub.md)`
Indent four spaces for a sub-chapter

`[Title]()`
Draft chapter with no file yet

`# Part Title`
Group the chapters that follow

`[Appendix](app.md)`
Suffix chapter, unnumbered, after the list

Use one list marker (- or *) throughout; mixing them breaks parsing.

BOOK.TOML

`[book] title = "<title>"`
Book title

`[book] authors = ["<name>"]`
Author list

`[book] src = "<dir>"`
Source directory (default src)

`[book] language = "<code>"`
Language for the HTML lang attribute

`[build] build-dir = "<dir>"`
Output directory (default book)

`[build] create-missing = false`
Fail on files missing from SUMMARY

BOOK.TOML (cont.)

`[rust] edition = "2021"`
Edition for Rust snippets (default 2015)

`[output.html] default-theme = "<name>"`
Pick the default colour theme

`[output.html] git-repository-url = "<url>"`
Add a repository link in the header

`[output.html] edit-url-template = "<url>"`
Per-page edit link, with {path}

`[output.html] site-url = "/<path>/"`
Base URL when hosted in a subpath

`[preprocessor.<name>]`
Enable a preprocessor

MARKDOWN AND CODE

`{{#include <file>}}`
Insert a whole file

`{{#include <file>:2:10}}`
Insert lines 2 to 10

`{{#include <file>:<anchor>}}`
Insert the part between ANCHOR markers

`{{#rustdoc_include <file>:2}}`
Show lines, keep the rest hidden for tests

MARKDOWN AND CODE (cont.)

`` ```rust,ignore ``
Highlight, but skip the test

`` ```rust,no_run ``
Compile in tests, never run

`` ```rust,editable ``
Make the snippet editable in the page

`` ```rust,noplayground ``
Remove the play button

`# let x = 1;`
Hide a Rust line (still compiled)

RECIPES

New book with live preview

```
mdbook init mybook --title "My Book"
cd mybook
mdbook serve --open
```

Test snippets against your crate

```
cargo build
mdbook test -L target/debug/deps
```

book.toml for a hosted repo

```
[book]
title = "Guide"
src = "src"
[rust]
edition = "2021"
[output.html]
git-repository-url = "https://github.com/<user>/<repo>"
site-url = "/<repo>/"
```

Build for CI

```
mdbook build -d public
mdbook test
```