



# Minimal Books

## Manual

Simple and complete books without introducing new syntax

---

typst

min-book\*

1.4.0

MIT-0

Maycon F. Melo<sup>†</sup>

## Contents

Quick Start .....	2
Description .....	2
Options .....	2
Advanced Configuration .....	4
Book Parts .....	5
Book Chapters .....	6
Theming & Customization .....	6
Cataloging in Publication .....	7
Commands .....	8
Note .....	8
Appendices .....	9
Annexes Command .....	10
Horizontal Rule .....	10
Block Quote Command .....	11
Copyright .....	11

---

\*<https://github.com/mayconfmelo/min-book>

<sup>†</sup><https://github.com/mayconfmelo>

# Quick Start

```
#import "@preview/min-book:1.4.0": book
#show: book.with(
  title: "Book Title",
  subtitle: "Book subtitle",
  authors: "Book Author",
)
```

## Description

Generate complete and complex books, without any annoying new commands or syntax, just good old pure Typst. This package manipulates the standard Typst elements as much as possible, adapting them to the needs of a book structure in a way that there's no need to learn a whole new semantic just because of *min-book*.

For some fancy book features there is no existing compatible Typst element to re-work and adapt; in those cases, this package do provide additional commands that are completely optional, for the sake of completeness.

This package comes with some thoughtful ready-to-use defaults but also allows you to play with highly customizable options if you need them, so it's really up to you: customize it your way or ride along the defaults — both ways are possible and encouraged.

## Options

```
#show: book.with(
  title: none,
  subtitle: none,
  edition: 1,
  volume: 0,
  authors: none,
  date: datetime.today(),
  cover: auto,
  titlepage: auto,
  catalog: none,
  errata: none,
```

```

dedication: none,
acknowledgements: none,
epigraph: none,
toc: true,
part: auto,
chapter: auto,
cfg: auto,
)

```

**title:** string content *(required)*  
Main title.

**subtitle:** string content none  
Subtitle, generally two lines long or less.

**edition:** integer  
Edition number (counts changes and updates between book releases).

**volume:** integer  
Series volume number (situates the book inside a collection).

**authors:** string array of strings *(required)*  
Author or authors.

**date:** datetime array dictionary  
(year, month, day)  
Publication date.

**cover:** auto function image content none  
Cover (overrides theme); when function, takes (meta, cfg) => { }  
and generate a back cover (use meta.is-back-cover) when enabled by  
#book(cfg.cover.back: true).

**titlepage:** auto content none  
Title page, shown after cover (overrides theme).

**catalog:** dictionary yaml toml  
Cataloging-in-publication board, used for library data (see “Cataloging in Publication” section).

**errata:** content string  
Correction of errors from previous book editions.

**dedication:** content string

Tribute or words addressed to someone important.

**acknowledgements:** content string

Expression of gratitude to those who contributed to the work.

**epigraph:** quote content

Brief quotation that foreshadows the book's theme, tone, or central reflection.

**toc:** boolean

Table of contents.

**part:** string none auto

Custom name of the book's main divisions (parts).

**chapter:** string none

Custom name of sections of the book (chapters).

**cfg:** dictionary

Set advanced configurations (see "Advanced Configuration") section.

## Advanced Configuration

```

cfg: (
  numbering: auto,
  transl: auto,
  std-toc: false,
  chapter-continuous: true,
  two-sided: true,
  paper-friendly: true,
  notes-page: false,
  theme: themes.stylish,
  styling: (:),
  cover: (:),
  part: (:),
)

```

These `#book(cfg)` configurations allows to modify certain aspects of the book and manage its appearance and structure. Built with some thoughtful ready-to-use defaults that make its use optional, so that beginners and casual writers can safely ignore it and *just write*.

**numbering:** string any

Heading numbering (managed by theme).

**transl:** string dictionary of strings

"file" (lang: "file")

Set custom Fluent files to *transl*<sup>1</sup> translation database (check `src/l10n/` for practical references).**std-toc:** booleanRestore default `#outline` appearance.**chapter-continuous:** boolean

Continue chapter (level 2) numbering even after a book part (level 1).

**two-sided:** boolean

Optimizes to print content on both sides of the paper.

**paper-friendly:** boolean

Use links attached to URL footnotes.

**notes-page:** booleanForces `#note` data to always appear in a separate new page.**theme:** module

Set book theme.

**styling:** dictionary

Theme general styling configurations.

**cover:** dictionary

Theme cover/title page configurations.

**part:** dictionary

Theme part configurations.

## Book Parts

```
#show: book.with(part: "Act")
= This is a part! // Act 1
```

---

<sup>1</sup><https://typst.app/universe/package/transl>

Some larger books are internally divided into multiple *parts*. This structure allows to better organize and understand a text with multiple sequential plots, or tales, or time jumps, or anything that internally differentiate parts of the story. Parts can also be called subjects, books, acts, units, modules, etc.; by default, *min-book* tries to use “Part” translated to `#text.lang` as name.

When set, all level 1 headings become *parts* and its appearance is managed by the current theme.

## Book Chapters

```
#show: book.with(chapter: "Scene")
== This is a chapter! // Scene 1
```

```
#show: book.with(chapter: "Scene", part: none)
= This is a chapter! // Scene 1
```

In most cases, books are divided into smaller sections called *chapters*. Generally, each chapter contains a single minor story, or event, or scene, or any type of subtle plot change. Chapters can also be called sections, articles, scenes, etc.; by default, *min-book* tries to use “Chapter” translated to `#text.lang` as name.

Chapters are smart: when set, if `#book(parts: none)` then all level 1 headings become chapters; otherwise, all level 2 headings become chapters — since parts are level 1 headings.

## Theming & Customization

Book customization relies on three pillars: themes, defaults, and settings. Themes are broader and more practical sets of customizations; defaults define pre-programmed adjustments that can be overridden; and settings allow for fine-tuning specific behaviors.

Built-in themes are provided under the `themes` module for quick visual customization; it is also possible to create a custom theme (check the `docs/themes.md` file for a practical reference). A theme is applied as following:

```
#import "@preview/min-manual:1.4.0": book, themes
#show: book.with( cfg: (theme: themes.default) )
```

Themes can provide defaults that can be overridden by `#set` rules defined before the `#show: book` command — defaults take precedence over rules after it.

```
#set page(margin: 2cm)
#set text(size: 18pt)
#set outline(depth: 4)

#show: book.with(...)
```

The settings are options of the `#book` command and allow for more specific adjustments. Some settings can be used by themes; they can also control defaults.

## Cataloging in Publication

```
catalog: (
  id: none,
  place: none,
  publisher: none,
  isbn: none,
  subjects: (),
  access: (),
  ddc: none,
  udc: none,
  before: none,
  after: none,
  bib-style: "associacao-brasileira-de-normas-tecnicas",
)
```

These `#book(catalog)` options set the data used to create the “cataloging in publication” board. Other needed information are automatically retrieved from the book data, but at least one of these options must be explicitly set to generate the board; otherwise it will be just ignored.

**id:** string content

*ACutter-Sanborn identification code*,<sup>2</sup> used to identify the book author.

**place:** string

The city or region where the book was published.

---

<sup>2</sup><http://www.cutternumber.com/>

**publisher:** string

The organization or person responsible for releasing the book.

**isbn:** string content

The *International Standard Book Number*, used to identify the book.

**subjects:** array of strings

A list of subjects related to the book.

**access:** array of strings

A list of access points used to find the book in catalogues, like by "Title" or "Series".

**ddc:** string content

*ADewey Decimal Classification*<sup>3</sup> number, which corresponds to the specific category of the book.

**udc:** string content

*AnUniversal Decimal Classification*<sup>4</sup> number, which corresponds to the specific category if the book.

**before:** content

Content showed before (above) the cataloging in publication board; generally shows editorial data like publisher, editors, reviewers, copyrights, etc.

**after:** content

Content showed after (below) the cataloging in publication board; generally shows additional information that complements the board data.

**bib-style:** string

Bibliographic reference style of the book data.

## Commands

### Note

```
#note(
  numbering: auto,
```

<sup>3</sup><https://www.oclc.org/content/dam/oclc/dewey/ddc23-summaries.pdf>

<sup>4</sup><https://udcsummary.info/php/index.php>



```
    body,
  )
```

Adds an end note, an alternative for footnotes but placed inside of page body instead of its footer. End notes generally appears in a separated page at the end of the current section, right before the next heading.

**numbering:** `auto` `string`

Custom note numbering (restarts note counter when set).

**body** `content`

*(required)*

The content of the end note.

## Appendices

```
#appendices(
  body,
  title: auto,
  numbering: (
    "",
    "{2:A}.\n",
    "{2:A}.{3:1}.",
    "{2:A}.{3:1}.{4:1}.",
    "{2:A}.{3:1}.{4:1}.{5:1}.",
    "{2:A}.{3:1}.{4:1}.{5:1}.{6:a}.",
  ),
  type: "appendix",
)
```

Creates an special ambient to write or include multiple appendices. An appendix is any important additional data left out of the main document for some reason, but directly referenced or needed by it. Inside this ambient, all level 1 heading is a new appendix.

**body** `content`

The appendices ambient.

**title:** `array of strings` `auto`

(singular, plural)

Title for each appendix and for Appendices section.

**numbering:** array of strings string

Custom appendices numbering (uses *numbly*<sup>5</sup> package).

## Annexes Command

```
#annexes(
  body,
  title: auto,
  numbering: (
    "",
    "{2:A}.\n",
    "{2:A}.{3:1}.",
    "{2:A}.{3:1}.{4:1}.",
    "{2:A}.{3:1}.{4:1}.{5:1}.",
    "{2:A}.{3:1}.{4:1}.{5:1}.{6:a}.",
  ),
)
```

Creates an special ambient to write or include multiple annexes. An annex is any important third-party data directly cited or referenced in the main document. Inside this ambient, all level 1 heading is a new annex.

**body** content

The annexes ambient.

**title:** array of strings auto

(singular, plural)

Title for each annex and for Annexes section.

**numbering:** array of strings string

Custom annexes numbering (uses *numbly*<sup>6</sup> package).

## Horizontal Rule

```
#horizontalrule()
```

Adds a horizontal rule, visual separators used to distinguish subtle changes of subject in extensive texts. Its appearance is controlled by the current theme.

---

<sup>5</sup><https://typst.app/universe/package/numbly>

<sup>6</sup><https://typst.app/universe/package/numbly>

The `#horizontalrule` command is also available as the smaller `#hr` alias.

## Block Quote Command

`#blockquote`(by: `none`, `..args`)

A simple alias for `#quote(block: true)` with a shorter attribution option.

## Copyright

Copyright © 2026 Maycon F. Melo.

This manual is licensed under MIT.

The manual source code is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

The logo was obtained from Flaticon website.