



Minimal Books

Manual

Simple and complete books without introducing new syntax.

min-book

0.1.1

MIT

Maycon F. Melo*

QUICK START

```
#import "@preview/min-book:0.1.1": book
#show: book.with(
  title: "Book Title",
  subtitle: "Complementary subtitle, not more than two lines long",
  authors: "Author",
)
```

DESCRIPTION

Generate complete and complex books, without any annoying new commands or syntax, just good old pure Typst. This package works by manipulating the standard Typst elements, adapting them to the needs of a book structure. All of this is managed behind the scenes, so that nothing changes in the Typst code itself.

While it is possible to play with complex structures, such as parts and chapters and creative numbering, this package comes with several ready to use default values; so its really up to you to customize it your way or ride along the defaults and just start writing: both ways are possible and encouraged.

This manual will be updated only when new versions break or modify something; otherwise, it will be valid to all newer versions starting by the one documented here.

*<https://github.com/mayconfmelo>

OPTIONS

Those are the full list of options available and its default values:

```
#import "@preview/min-book:0.1.1": book
#show: book.with(
  title: none,
  subtitle: none,
  authors: none,
  date: datetime.today(),
  cover: auto,
  titlepage: false,
  part: "Part",
  chapter: "Chapter",
  numbering-style: auto,
  toc: true,
  paper: "a5",
  lang: "en",
  justify: true,
  line-space: 0.5em,
  par-margin: 0.75em,
  first-line-indent: 1em,
  margin: (x: 15%, y: 14%),
  font: ("Book Antiqua", "Times New Roman"),
  font-math: "Asana Math",
  font-size: 11pt,
)
```

Seems like an awful lot to start with, but let's just break down all this to understand it better, shall we?

title: `string` `content` *(required)*

The main title of the book.

subtitle: `string` `content` `none`

The book subtitle; generally two lines long or less.

authors: `string` `array` *(required)*

A string or array containing the name of each author of the book, in format (NAME, NAME).

date: array auto

The book publication date, in format (yyyy, mm, dd). Fallback to current date if not set, or set to auto.

cover: auto content none

The book cover; automatically generates a default cover, or set a custom content as cover.

titlepage: boolean content

The page after the cover; by default, contains the title, subtitle, author, and year of publication. It is possible to create the titlepage using a content block.

part: string none

The name given to the book's main divisions — something like LaTeX's `\part` command.

chapter: string none

The name given to the book's main sections — something like LaTeX's `\chapter` command.

numbering-style: auto array string none

Defines a custom heading numbering. Can be a standard numbering string, or a *numbly*¹ numbering array in more complex cases.

toc: boolean

Defines whether the book will have a table of contents or not.

paper: string dictionary

Defines the page paper type, or its custom size; a dictionary (x: LENGTH, y: LENGTH) can be used to set custom sizes for page width (x) and height (y).

lang: string

Defines the language of the written text.

justify: boolean

Defines if the text will have justified alignment.

¹<https://typst.app/universe/package/numbly>

line-space: length

Defines the space between lines in the document.

par-margin: length

Defines the margin space after each paragraph. Set it the same as `line-space` to remove get paragraphs without additional space in between.

first-line-indent: length

Defines the first line indentation of all paragraphs, but the first one, in a sequence of paragraphs.

margin: length

Defines the document margins.

font: string array

Defines the font families used for the text: a principal font and its fallback.

font-math: string array

Defines the font families used for the math and equations: a principal font and its fallback.

font-size: length

Defines the size of the text in the document.

BOOK PARTS

Some larger books are internally divided into multiple *parts*. This feature 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. While the name used here is *part*, different books uses different names for it: parts, subjects, books, acts, etc.

Parts are used in a smart way: when enabled, every level 1 headings are turned into *book parts* automatically. To enable, just give a string value to the `part` argument; and to disable it, set to `none`:

```
// Enable part with name "Act":
#show: book.with(
  part: "Act",
)
```

= This is Act 1

```
// Disable part to turn it back to normal.
#show: book.with(
  part: none,
)
```

= This is a normal level 1 heading

BOOK CHAPTERS

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 are used in a smart way, too, depending of the use of *book parts*: if *parts* are enabled, every level 2 headings are turned into *book chapters*; but if *parts* are disabled, all level 1 headings are turned into *book chapters*. Seems complex, but it turns to be very intuitive:

```
// Book with parts and chapters:
#show: book.with(
  part: "Act",
  chapter: "Scene",
)
```

= This is Act 1

== This is Scene 1

```
// Book with just chapters:
#show: book.with(
  part: none,
  chapter: "Scene"
)
```

= This is Scene 1

```
// Boom with no part, nor chapter:
#show: book.with(
  part: none,
  chapter: none,
)
```

= This is a level 1 heading

== This is a level 2 heading

ADVANCED NUMBERING

The book headings can be numbered two ways: using a standard numbering string, or a *numbly*² numbering array. While numbering strings are indicated for simpler cases, the numbly arrays are used in more complex book numbering.

By default, when the book have *parts* enabled, the following numbering is used:

```
(
  "{1:I}:\n",
  "{2:I}.\n",
  "{2:I}.{3:1}.\n",
  "{2:I}.{3:1}.{4:1}.\n",
  "{2:I}.{3:1}.{4:1}.{5:1}.\n",
  "{2:I}.{3:1}.{4:1}.{5:1}.{6:a}.\n"
)
```

But when the *parts* are disabled, the following numbering is used:

```
(
  "{1:I}.\n",
  "{1:I}.{2:1}.\n",
  "{1:I}.{2:1}.{3:1}.\n",
  "{1:I}.{2:1}.{3:1}.{4:1}.\n",
  "{1:I}.{2:1}.{3:1}.{4:1}.{5:1}.\n",
  "{1:I}.{2:1}.{3:1}.{4:1}.{5:1}.{6:a}.\n"
)
```

Additionally, a `numbering-style` argument can be set to override these default numbering above.

ADDITIONAL FEATURES

As said before, this package does not introduce any new syntax nor commands to write a complete book. But, well... it does offers some additional commands. These are completely optional and are available only by explicit import, separate from the `#book` main function:

NOTE COMMAND

Adds end notes to the book. End notes are more common in books than footnotes, and while footnotes appear at the footer of the same page, end notes appears at its own page at the end of the current section, right before the next heading:

²<https://typst.app/universe/package/numbly>

```
#import "@preview/min-book:0.1.1": note
```

= Current heading

This text have an end note#note[This is an end note]

```
// The end note will automatically appear here.
```

= Next heading

HORIZONTAL RULE COMMAND

Adds horizontal rules, used to separate subtle changes of subject in book texts. Can be called by its name or its alias:

```
#import "@preview/min-book:0.1.1": horizontalrule, hr
```

This text have a subject.

```
#horizontalrule()
```

This text have another subject.

```
#hr()
```

And this text have some other subject.

This command have the following optional arguments:

symbol: [**#sym.ast.op** **#sym.ast.op** **#sym.ast.op**] **content**

Defines the content at the center of the horizontal rule. By default, its line is struck by three * in its center.

spacing: **1em** **length**

Defines the vertical space before and after the horizontal rule.

line-size: **15%** **length**

Defines the size of the horizontal rule line.

BLOCK QUOTE COMMAND

Adds a block version of the quote command. In fact, it is just a simple wrapper of `quote(block: true)` with some modifications:

```
#import "@local/min-book:0.1.1": blockquote

#blockquote(by: "Einstein")[
  Don't believe everything you read on the internet.
]
```

APPENDICES COMMAND

Creates an special ambient to write or include multiple appendices, with special heading styling.

```
#import "@local/min-book:0.1.1": appendices

#appendices[
  #include("appendix-a.typ")
  #include("appendix-b.typ")
  #include("appendix-c.typ")
]
```

Appendices are important additional data left out of the main document for some reason, but directly cited or referenced by it.

ANNEXES COMMAND

Creates an special ambient to write or include multiple annexes, with special heading styling.

```
#import "@local/min-book:0.1.1": annexes

#annexes[
  #include("appendix-a.typ")
  #include("appendix-b.typ")
  #include("appendix-c.typ")
]
```

Annexes are important third-party data directly cited or referenced in the main document.

COPYRIGHT

Copyright © 2025 Maycon F. Melo.

This manual is licensed under MIT terms and rights.

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.