

test pagina

Lars Devocht

test pagina

Lars Devocht

This book is for sale at <http://leanpub.com/lars>

This version was published on 2023-12-06



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.

© 2023 Lars Devocht

Contents

Dependency tracking	1
Backward tracking : dependencies between effects and reactive properties	1
Example that implements forward tracking	1
Writing in Markua	3
Section One	3
Including a Chapter in the Sample Book	3
Links	3
Images	3
Lists	8
Code Samples	9
Tables	9
Math	10
Headings	10
Block quotes, Asides and Blurbs	11
Good luck, have fun!	13

Dependency tracking

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Forward tracking: from reactive properties to effects

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Code snippets explanation

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Conceptual Visualization

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Backward tracking : dependencies between effects and reactive properties

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Example that implements forward tracking

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

activeEffect

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

targetMap

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

track(target, key)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

trigger(target, key)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

ReactiveEffect class

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

effect(fn)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

reactive(obj)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/lars>

Writing in Markua

Writing in Markua is easy! You can learn most of what you need to know with just a few examples.

To make *italic text* you surround it with single asterisks. To make **bold text** you surround it with double asterisks.

Section One

You can start new sections by starting a line with two # signs and a space, and then typing your section title.

Sub-Section One

You can start new sub-sections by starting a line with three # signs and a space, and then typing your sub-section title.

Including a Chapter in the Sample Book

At the top of this file, you will also see a line at the top:

```
{sample: true}
```

Leanpub has the ability to make a sample book, which interested readers can download or read online. If you add this line above a chapter heading, then when you publish your book, this chapter will be included in a separate sample book for these interested readers.

Links

You can add web links easily.

Here's a link to the [Leanpub homepage](https://leanpub.com)¹.

¹<https://leanpub.com>

Images

You can add an image to your book in a similar way.

First, add the image to the “Resources” folder for your book. You will find the “Resources” folder under the “Manuscript” menu to the left.

If you look in your book’s “Resources” folder right now, you will see that there is an example image there with the file name “palm-trees.jpg”. Here’s how you can add this image to your book:



If you want to add a figure title, you put it in quotes:



Figure 1. Palm Trees

If you want to add descriptive alt text, which is good for accessibility, you put it between the square brackets:



Figure 2. Palm Trees

You can also set the alt text and/or the figure title in an attribute list:



Figure 3. Palm Trees

Finally, if no title is provided, and the `alt-title` document setting is the default of `all`, the alt text will be used as the figure title instead of as alt text.



Figure 4. Palm Trees

You can set the important document settings at Settings > Generation Settings.

Lists

Numbered Lists

You make a numbered list like this:

1. kale
2. carrot
3. ginger

Bulleted Lists

You make a bulleted list like this:

- kale
- carrot
- ginger

Definition Lists

You can even have definition lists!

term 1

definition 1a

definition 1b

term 2

definition 2

Code Samples

You can add code samples really easily. Code can be in separate files (a “local” resource) or in the manuscript itself (an “inline” resource).

Local Code Samples

Here’s a local code resource:

Figure 5. Hello World in Ruby

```
puts "hello world"
```

Inline Code Samples

Inline code samples can either be spans or figures.

A span looks like `puts "hello world"` this.

A figure looks like this:

```
puts "hello"
```

You can also add a figure title using the title attribute:

Figure 6. Hello World in Ruby

```
puts "hello"
```

Tables

You can insert tables easily inline, using the GitHub Flavored Markdown (GFM) table syntax:

Header 1	Header 2
Content 1	Content 2
Content 3	Content 4 Can be Different Length

Tables work best for numeric tabular data involving a small number of columns containing small numbers:

Central Bank	Rate
JPY	-0.10%
EUR	0.00%
USD	0.00%
CAD	0.25%

Definition lists are preferred to tables for most use cases, since reading a large table with many columns is terrible on phones and since typing text in a table quickly gets annoying.

Math

You can easily insert math equations inline using either spans or figures.

Here's one of the kinematic equations $d = v_i t + \frac{1}{2} a t^2$ inserted as a span inside a sentence.

Here's some math inserted as a figure.

$$\left| \sum_{i=1}^n a_i b_i \right| \leq \left(\sum_{i=1}^n a_i^2 \right)^{1/2} \left(\sum_{i=1}^n b_i^2 \right)^{1/2}$$

Figure 7. Something Involving Sums

Headings

Markua supports both of Markdown's heading styles.

The preferred style, called atx headers, has the following meaning in Markua:

```
{class: part}
# Part
```

This is a paragraph.

```
# Chapter
```

This is a paragraph.

```
## Section
```

This is a paragraph.

```
### Sub-section
```

This is a paragraph.

```
#### Sub-sub-section
```

This is a paragraph.

```
##### Sub-sub-sub-section
```

This is a paragraph.

```
##### Sub-sub-sub-sub-section
```

This is a paragraph.

Note the use of three backticks in the above example, to treat the Markua like inline code (instead of actually like headers).

The other style of headers, called Setext headers, has the following headings:

```
{class: part}
Part
====
```

This is a paragraph.

```
Chapter
=====
```

This is a paragraph.

```
Section
-----
```

This is a paragraph.

Setext headers look nice, but only if you're only using chapters and sections. If you want to add sub-sections (or lower), you'll be using atx headers for at least some of your headers. My advice is to just use atx headers all the time. (The `{class: part}` attribute list on a chapter header to make a part header does actually work with Setext headers, but it's really ugly.)

Note that while it is confusing and ugly to mix and match using atx and Setext headers for chapters and sections in the same document, you can do it. However, please don't.

Block quotes, Asides and Blurbs

Block quotes are really easy too.

—Peter Armstrong, *Markua Spec*

Asides are useful for longer text.

But typing them like this isn't fun.

Asides can be written this way, since adding a bunch of A> stuff at the beginning of each line can get annoying with longer asides.

Blurbs are useful

Blurbs are useful

There are many types of blurbs, which will be familiar to you if you've ever read a computer programming book.



This is a discussion.

You can also specify them this way:



This is a discussion



This is an error.



This is information.



This is a question. (Not a question in a Markua course; those are done differently!)



This is a tip.



This is a warning.



This is an exercise. (Not an exercise in a Markua course; those are done differently!)

Good luck, have fun!

If you've read this far, you're definitely the right type of person to be here!

Our last piece of advice is simple: once you have a couple chapters completed, publish your book in-progress!

This approach is called Lean Publishing. It's why Leanpub is called Leanpub.

If you want to learn more about Lean Publishing, read [this](#)² or watch [this](#)³.

²<https://leanpub.com/lean/read>

³<https://youtu.be/ozO0kOnqmyA>