

# Example Markdown Configuration for Pandoc

## Contents

<b>1</b>	<b>Pandoc</b>	<b>1</b>
1.1	Installation MacOS . . . . .	1
1.2	Frontmatter . . . . .	1
1.3	Pandoc Arguments . . . . .	2
1.4	Simple Table Format for Pandoc . . . . .	2
1.5	Pandoc Lua Filters . . . . .	2
1.6	Shared Options . . . . .	3
1.7	Nested Headings . . . . .	3
1.7.1	Third Level . . . . .	3

## 1 Pandoc

See the *source code of this Markdown file* to see the configuration settings. **Die Reihenfolge im YAML-Block ist sehr wichtig!**

### 1.1 Installation MacOS

```
1 # Install pandoc
2 brew install pandoc
3 # Install basictex (It is a small TeX Live distribution)
4 brew install basictex
5 # Install tlmgr (TeX Live Manager)
6 wget https://mirror.ctan.org/systems/texlive/tlnet/update-tlmgr-latest.sh
7 sudo sh ./update-tlmgr-latest.sh
8 # See installed packages
9 tlmgr list --only-installed
10 # Install the missing packages
11 sudo tlmgr install xetex # for xelatex
12 sudo tlmgr install framed # for boxes
13 sudo tlmgr install soul # for highlighting
14 sudo tlmgr install fvextra # for fancyvrb
15 sudo tlmgr install mdframed
16 sudo tlmgr install zref
17 sudo tlmgr install needspace
18 sudo tlmgr install titlesec
19
```

### 1.2 Frontmatter

- Pandoc Frontmatter
  - <https://pandoc.org/MANUAL.html#metadata-blocks>
- Pandoc Variables
  - <https://pandoc.org/MANUAL.html#variables-for-latex>
- Pandoc Arguments
  - <https://shd101wyy.github.io/markdown-preview-enhanced/#/pandoc-word?id=pandoc-arguments>
- Pandoc Syntax Highlighting
  - <https://pandoc.org/MANUAL.html#syntax-highlighting>
- Pandoc Word
  - <https://shd101wyy.github.io/markdown-preview-enhanced/#/pandoc-word?id=pandoc-arguments>

- Pandoc Highlighting Themes
  - *tango*, *pygments*, *kate*, *monochrome*, *espresso*, *zenburn*, *haddock*, *breezedark*, *breezelight*, *textmate*, and *zenburn*

**TOC** is the table of contents.

But it is better to generate `toc` by `markdown-preview-enhanced`, because you can set here not only the depts of the `toc`, but also the starting level and the end level, e.g. `depthFrom=2 depthTo=4`.

However, **Pandoc** can **number the chapters and sub-chapters better**, something that `markdown-preview-enhanced` doesn't do as well.

If you want to create a print document, use `pandoc's toc`, if you want to read at the internet, use `markdown-preview-enhanced's toc`. In this file, both are used.

### 1.3 Pandoc Arguments

If there are `pandoc` features you want to use that lack equivalents in the `YAML` options described above you can still use them by passing custom `pandoc_args`. For example:

```
1 ---
2 title: "Habits"
3 output:
4   word_document:
5     pandoc_args: ["--cs1", "/var/csl/acs-nano.csl"]
6 ---
```

### 1.4 Simple Table Format for Pandoc

**Prettier** has still problems with **markdown-linters in VSCode**. If you use `prettier`, it changes its formatting and table is broken. Therefore use `markdownlint` instead and change the settings for `markdown` in **VSCode** settings to:

```
1 "[markdown]": {
2   "editor.defaultFormatter": "DavidAnson.vscode-markdownlint",
3   "editor.formatOnSave": true,
4 },
```

To be absolutely sure, that `prettier` ignores all `markdown` files, add the following to your `.prettierrignore` file:

```
1 *.md
```

So finally, you can use this beautiful `pandoc` table format (You have to set **pandoc** as your **default markdown renderer** in `markdown-preview-enhanced`):

```
1 "markdown-preview-enhanced.enableCriticMarkupSyntax": true,
2 "markdown-preview-enhanced.enableExtendedTableSyntax": true,
3 "markdown-preview-enhanced.usePandocParser": true,
```

Centered Header	Default Aligned	Right Aligned	Left Aligned
First	row	12.0	Example of a row that spans multiple lines. Here's another one. Note the blank line between rows.
Second	row	5.0	

**GitHub** cannot render the table correctly.

### 1.5 Pandoc Lua Filters

`Pandoc Lua Filters` are a powerful way to extend `Pandoc`. They are written in `Lua` and can be used to modify the abstract syntax tree (AST) that `Pandoc` uses to represent the document being converted. This allows you to customize look and feel, add metadata, and perform many other functions. `Lua` filters can be used with all of

Pandoc's input formats (Markdown, reStructuredText, HTML, LaTeX, etc.), and can produce any of its output formats (including native Haskell formats).

...

See [Pandoc Lua Filters](#) for more details.

## 1.6 Shared Options

See <https://shd101wyy.github.io/markdown-preview-enhanced/#/pandoc-word?id=shared-options>

If you want to specify a set of default options to be shared by multiple documents within a directory you can include a file named `_output.yaml` within the directory. Note that no YAML delimiters or enclosing output object are used in this file. For example:

**`_output.yaml`**

```
word_document:
  highlight: zenburn
```

All documents located in the same directory as `_output.yaml` will inherit it's options. Options defined explicitly within documents will override those specified in the shared options file.

## 1.7 Nested Headings

### 1.7.1 Third Level

Some third level text.

**1.7.1.1 Fourth Level** Some fourth level text.

**1.7.1.1.1 Fifth Level** Some fifth level text.