Introduction

The goal of this document is to demonstrate how easy it is to generate beautiful documents with just a little knowledge of Markdown and a tool called Pandoc. I want to debunk the myth that **Docs as Code** (or **DaC** for short) is ...

- ... hard to learn.
- ... only used to document software projects.
- ... only suitable for programmers and/or engineers.
- ... would require you to learn a lot of LaTeX.

Always remember:

> Documents as Code is easy!

This document is not supposed to do so by itself. It is not some guide or tutorial. The purpose of this document is to have something to work with to demonstrate different aspects of the **DaC** approach to generate nice looking documents. I set up a website that explains everything and this document will be used in the explanation.

It is also not a manual for Pandoc as this already exists online or a Markdown guide, because there are already plenty of resources available online that explain the Markdown syntax, e.g. this one.

My plan is to set up a demonstration that is split into four levels (from *beginner* to *pro*). The higher the level the better the generated document will look. Nevertheless, only the highest two levels will require any LaTeX knowledge. But even at the second level, the document will already look really professional and most likely way better than anything you create with Word.

1. **Beginner**:

   Let's start simple. Just use a plain Markdown file and the simplest form of the `pandoc` command to build the output PDF file: no additional options/parameters, just the bare minimum.

   Assuming the source Markdown file is called `solution_beginner.md`, the following command renders the PDF output file `solution_beginner.pdf`: `pandoc -o solution_beginner.pdf solution_beginner.md`

2. **Intermediate**:

   Here we ...

   - add a first additional option to the `pandoc` command. This option enumerates the sections.
   - add some front matter with additional information about the document.
   - add some additional metadata to configure the styling of the output. This drastically improves the look of the output.

The output are three consecutively better looking output files.

3. **Advanced**:

   This is the first time, a little bit of LaTeX syntax is used to further improve the look of the PDF file. But don't worry, it is very simple and everything is explained. After that you should be able to make changes in the given example to tweak the look further to your liking.

   Pandoc can be extended by the use of filters. I show how to use one of them as an example. This will for instance allow us to reference images (see the Figures section or the Inline Image section). See [chapter @sec:inline-images].

   The **advanced** level is split into two steps (two output files).

4. **Pro**:

   For the **Pro** level I figured it is time to write a little LaTeX template for Pandoc to get the maximum out of it.

All right let's get started. The following chapters won't necessarily contain meaningful content. Their main purpose is to demonstrate different possibilities for formatting and other features.

Images

Images can either be added as figures or inline. When added as a figure, a caption will be added that include the current number of the figure which is automatically increase with every figure. When added inline, all that is added is an image, nothing else.

Figures

Figures are *floating* objects in LaTeX. Since Pandoc used LaTeX in the background, figures might not appear where you expect them to appear. While this might be annoying in the beginning, this is actually a feature. Also you should always refer to an included figure somewhere in the text. If you don't do that, you might as well just not add the figure.

[Figure @fig:example] shows a section on the way between Santiago de Chile and Mendoza in Argentina.

Inline Images

With inline images, you can for example place two images next to each other:



Figure 1: Sample Figure



Math

Something that is often needed is math expressions/equations/formulas. Just like images math can be inserted inline or as separate entities.

Inline Math

One of the most famous equations of all time is probably $E = mc^2$.[1]

Equations

Writing the equation from the previous section inline, like I did, doesn't really do it justice. Therefore it is also possible to render it a little more noticeable like this:

[1] According to Wikipedia Einstein was the first one to propose the equivalence of mass and energy.

$$E = mc^2$$

{#eq:mass_energy}

[Equation @eq:mass_energy] looks way better than the inline version doesn't it?[2]

If you want, you can use superscripts and subscripts in normal text as well. Here are two examples: H_2O and 2^{10}.

By the way, Albert Einstein published a bunch of papers. One is called "Zur Elektrodynamik bewegter Körper" [see @einstein1905electrodynamik, pp. 42].

Tables

Last but not least, let's add a table to the document. This can be done in multiple ways. Please read Pandoc's User Guide for more information. I will just add one table that uses the most complicated type. I wanted to use that type to demonstrate that you can create pretty complicated tables with cells that span multiple rows and/or columns. Admittedly the Markdown code is a little bit cumbersome, but if you don't need all the features shown here, there are simpler table types.

Table 1: Table Example {#tbl:example_table}

Fruit		Prize
Apple	green round	1.00
	red round	1.25
Bananas	yellow curved	0.50
Total		2.75

To add [table @tbl:example_table], I had to add the following to the Markdown source file:

```
: Table Example

+---------------+---------------+------------+
| Fruit                         | Prize      |
```

[2] Of course it does. Who questioned it?

```
+:==============+:=============:+===========:+
| Apple          |  green \      | 1.00       |
|                |  round \      |            |
|                +---------------+------------+
|                |  red \        | 1.25       |
|                |  round \      |            |
+----------------+---------------+------------+
| Bananas        |  yellow \     | 0.50       |
|                |  curved       |            |
+================+===============+============+
| Total          |               | 2.75       |
+================+===============+============+
```

ToDos

What is left to create this demonstration:

- ☒ ~~Set up the solution markdown source file.~~
- ☒ ~~Set up CMake to build the output PDF file.~~
- ☐ Write the accompanying tutorial.
- ☐ Actually put the website online.
- ☐ What else?

Feedback

If you have any form of feedback about this document or the accompanying website, don't hesitate to send me an email. The address is: contact@schuam.de. Or visit my website at https://www.schuam.de.