DOCUMENT TITLE

Gerard Keane

April 9, 2019

Contents

This project serves as a template to be used for creating documents. It's for my own personal use, including styling that I prefer, but anyone is free to use this. Whatever the least restrictive terms available are—that's what this repository is covered by. Fill your boots.

Requirements

- Pandoc[1]
- LaTeX [2] Specifically, I am using xelatex as the pdf-engine to use system fonts.

Outline

You can overwrite the text in this document using Markdown[3], which is extremely simple. View the documentation for more detail on how to use Markdown.

The beauty of this setup is that, thanks to pandoc, you can use Markdown and LaTeX in the same document to produce beautiful documents incredibly easily.

You should check out the metadata first. There is a separate yaml file called metadata.yaml which contains a few variables. You can leave it alone if you want obviously. Afterwards, simply write in template.md whatever it is you want to write. Then, run make in your command line and—hey presto—you will find a file called output.pdf in the same folder.

Document Metadata

```
author: Gerard Keane
title: Document Title
date: \today
```

Hopefully these are self-explanatory. If you want, you can write dates in the following format:

```
date: 9^th^ April 2019
```

Which will produce a date like this:

9th April 2019.

If you leave it alone you will get today's date in the format:

April 9, 2019.

[1] https://pandoc.org/
[2] https://www.latex-project.org/
[3] https://daringfireball.net/projects/markdown/

Table of Contents

The table of contents (*toc*), list of tables (*lot*) and list of figures (*lof*) can be automatically generated by using the following lines:

```
lot: false
toc: true
lof: false
```

They will be on a separate page from the title thanks to the following line in the YAML header, which you shouldn't touch.

```
include-before:
- '`\newpage{}`{=latex}'
```

Typography

The fonts used by the document are controlloed by the following variables:

```
mainfont: "Hoefler Text"
sansfont: "Helvetica Neue"
monofont: "Fira Code"
```

MAINFONT represents the body text of the document. SANSFONT is used for the section headers throughout, and MONOFONT is used for representing code, such as the blocks you can see in this section.

You can use any font installed on your system for these.

The fontsize is obviously set by:

```
fontsize: 12pt
```

Margins

The document is set to have 3cm margins all around the document. Set by the following line:

```
geometry: [top=3cm, bottom=3cm, left=3cm, right=3cm]
```

LaTeX Magic

The few minor alterations I have made to the default pandoc template are included in the section HEADER-INCLUDES, as below:

```
header-includes:
  # Font setup
  - \newfontfamily\headingfont[]{Gill Sans}
  - \newfontfamily\dropcapfont[]{Helvetica Neue}
```

```
# Setting The Title Styling to Gill Sans
- \usepackage{titling}
- \pretitle{\begin{center}\bfseries\headingfont\Huge\MakeUppercase}
- \posttitle{\end{center}\vspace{2em}}
# Setting the section heading styling to Helvetica
- \usepackage{sectsty}
- \allsectionsfont{\sffamily}
# For Drop Caps / Lettrines
- \usepackage{lettrine}
- \renewcommand{\LettrineFontHook}{\dropcapfont\bfseries}
# Lorem Ipsum Generator for helping layout document
- \usepackage{lipsum}
```

Since pandoc uses LaTeX to generate pdfs, one can use any LaTeX package in the manner described above. This is an extremely powerful tool and I urge you to examine the possibilities of this if you are in any way interested in further customising this document. There's a whole world out there.

I have set the title to use Gill Sans, and ensured that it is all uppercase. I have also set the section headings to use the sans-serif font as per the TYPOGRAPHY section above.

You can alter these however you like, I simply prefer the aesthetic of sans-serif headers and serif body text.

Examples

Mathematics

Input:

```
$$
\frac{sales - costs}{costs} \times 100 = markup \%
$$
```

Output:

$$\frac{sales - costs}{costs} \times 100 = markup\%$$

Sections

Sections are dealt with like this:

```
# Section
## Subsection
### Subsubsection
```

And so on up to 6 levels deep.

Lists

Unordered Lists

Input:

```
- First Item
- Second Item
- Third Item
```

Output:

- First Item
- Second Item
- Third Item

Ordered Lists

Input:

```
1. First Item
2. Second Item
3. Third Item
```

Output:

1. First Item
2. Second Item
3. Third Item

Text Emphasis

Input:

```
This is *italicised text*
```

Output:

This is *italicised text*

Input:

```
This is **bold text**
```

Output:

This is **bold text**

Tables

Input:

```
Table: Example Table

     Sales      Costs       Profit    %
---------- ---------- ------------ ----
556,415.42 701,645.04 (145,229.62)  -26
```

Output:

Table 1: Example Table

Sales	Costs	Profit	%
556,415.42	701,645.04	(145,229.62)	-26

Drop Caps & Lettrines

Drop caps and lettrines are more complicated to use, as they are not covered by pandoc markdown. But since we are using LaTeX to produce the pdf, we can use LaTeX classes throughout our document. Consider the following:

```
\lettrine[lines=9, findent=-3em, nindent=0.2em, slope=0.5em]{A} ll[l your base are belong to us]{•smallcaps}. \lipsum[1]
```

This code specifies a *Drop Cap* 9 lines deep, with the first line brought closer to the letter by -3em and the following lines pushed away from the letter 0.2em, sloping away by 0.5em. Some *Lorem Ipsum* is generated to demonstrate. Like this:

All your base are belong to us. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Experiment away with this, as you have an extraordinary amount of control over the appearance of this, even without editing metadata.yaml. You can view the documentation for the `\lettrine` package here[4]. Be warned though that not everything will necesarily work through pandoc, so test carefully. For instance, you are *supposed* to use it like this:

```
\lettrine[lines=9]{A}{ll your base are belong to us}. Lorem ipsum ...
```

But instead, thanks to some unknown vagary, we are forced to do this:

```
\lettrine[lines=9]{A} ll[l your base are belong to us]{•smallcaps}.
Lorem ipsum ...
```

I don't know why, and I've spent long enough trying to work it out. Suffice it to say that this works, and let's leave it at that.

[4] http://ctan.mirrors.hoobly.com/macros/latex/contrib/lettrine/doc/lettrine.pdf