Overview

I needed to write some documentation for something I developed at work.
Writing in markdown, and converting with Pandoc has been a really nice solution.

Issue

We used network drives to store information (plans, documentation, etc.) in selected team and project folders. I needed to quickly and easily create a bunch of different documentation to store there.

Until now, most of the documentation was done in Microsoft Word, which isn’t bad for documents intended to be printed, but there were a few drawbacks here:

1. Code formatting: I’m not sure it even exists, outside of a monospace font.
2. Jump to Sections: There is a sections area, but it behaves a bit janky.
3. Layout: Unless you go tearing up the layout settings, the page is very narrow, and line breaks act awkwardly.

Some of these things can (probably) be fixed on a document-by-document basis, but it’s not ideal to have to remembering which ones to set when creating a new document.

The other issue I’m sure you’ve all seen: Someone creates a beautiful looking Word document, everyone loves it. Then someone else comes along to maintain it, who isn’t a Word-Wizard, and the document slowly mutates into this hideous hybrid mess of the old ‘beautiful’ designed elements, the cheap knock-off elements the maintainer came up with that don’t quite match, and once they gave up: the default Word elements.

Finally, the documentation needed to be used on Windows 10.

So our documentation requirements are:

1. To live as files on a network drive.
2. To be (easily) maintainable by other people (including style).
3. Code formatting.
4. Windows compatible.

Resolution

If you hadn’t noticed already, even with all its flaws I really love Markdown (this blog is written in Markdown!), and luckily for me: in this case, it seemed very appropriate.

Using pandoc, it’s easy to convert Markdown formatted documents into HTML, PDF, or other formats.

1. The source and output files can live on the network drive.
2. Anybody can edit a text document!
3. Markdown (in particular Github Flavored Markdown) has great code formatting support!
4. Pandoc has a Windows binary

Great, everything is a match!

Write a Markdown Document

So here’s our document:

~~~
# How to eat carrots


1. Wash them
2. Peel them
3. Bite them
4. Chew them


# Troubleshooting
## Unable to bite
Make sure you have teeth.


## Unable to peel
You probably don't have a peeler, go buy one.


## You didn't write any code
```python
for i in range(0, 12):
    print("Hello World!")
    ```
~~~

Convert to HTML

Converting to HTML is as easy as running:

```
pandoc document.md -f gfm -t html -o my_document.html
```

Looking at that HTML page in your browser, it does look a bit glum. Let’s fix that.

Add some Styling

Killercup has saved us a bunch of time and created a CSS file we can use with our documents.

First, grab the .css file from here: pandoc.css, and put it in the same folder as your Markdown document.

Now, run this to generate your beautiful HTML documents:

```
pandoc document.md -f gfm -t html -o my_document.html --css pandoc.css
```

Great! That looks a lot better!

Convert to PDF

If you prefer PDF documentation, download wkhtmltopdf to the same folder as your Markdown document, then run this:

```
pandoc document.md -f gfm --pdf-engine=wkhtmltopdf.exe -o my_document.html --css pandoc.css
```

Now we have fancy looking PDFs from an easily editable text file!

Final Touches

I made some additional changes, like adding a Table of contents, and creating a couple of .bat scripts which I’ll share below. Keep in mind my folder structure was something like this:

```
project/
|   document1.pdf
|   document2.pdf
└───source/
    |     document1.md
    |     document2.md
    |     pandoc.css
    |     pandoc.exe
    |     wkhtmltopdf.exe
```

Markdown to HTML (Batch Script)

```
@echo off

echo "Converting..."

for %%F in (*.md) do pandoc %%~nxF -f gfm --pdf-engine=wkhtmltopdf.exe -o ../%%~nF.pdf

pause
```

Markdown to PDF (Batch Script)

```
@echo off

echo "Converting..."

for %%F in (*.md) do pandoc %%~nxF -f gfm -t html -s -o ../%%~nF.html --css source/pand

pause
```

As always, any issues: feel free to create a GitHub issue or a pull request.

Thanks,

l

github