

Markdown to PDF

Sublime setup to quickly convert Markdown files to PDF through LaTeX with Pandoc

Why

As a fairly new OSX user with my fancy new Macbook I came across a fair amount of issues that held me from optimizing my workflow. Since I have been using Windows for literally forever, not having the default Microsoft Office Word which I was used to on my Desktop was an issue. So I had to look for a solution where I could make documentation or reports for school without having to do a lot of effort.

Sublime

Sublime Text is a sophisticated text editor for code, markup and prose. You'll love the slick user interface, extraordinary features and amazing performance.

First of all, before we start all of this, you will need to have Sublime Text installed. So go ahead and go download Sublime Text (3) from: <https://www.sublimetext.com/3>

Sublime Package Control (optional)

For a quick and easy way to setup plugins and other handy things for Sublime install Sublime Package Control. This is optional but comes in handy if you'd like to preview your markdown files in your browser.

<https://packagecontrol.io/>

After installing Sublime Package Control you press `cmd+shift+p`. Search for Markdown Preview and click it, this will install a handy tool for you to preview your markdown files in your browser without much effort. To use simply press `cmd+shift+p` again and type in Markdown, select Markdown Preview and select browser.

Pandoc

If you need to convert files from one markup format into another, pandoc is your swiss-army knife. Pandoc can convert documents in markdown, reStructuredText, textile, HTML, DocBook, LaTeX, MediaWiki markup, TWiki markup, OPML, Emacs Org-Mode, Txt2Tags, Microsoft Word docx, LibreOffice ODT, EPUB, or Haddock markup.

You download Pandoc from here: <http://pandoc.org/installing.html>
You will need Pandoc to convert your Markdown files to PDF files.

Heck, you might even want to try and search for Pandoc in the Sublime Package Control and install the plugin.

This should allow you to convert your files as well. Since this wasn't working for me at all and since I really couldn't get my head around it, I decided to make a custom build option for Sublime.

Custom Sublime Build

First off we need to create a new Sublime-Build. Go to **Tools > Build System > New Build System ...**

This opens a file, replace the content with:

```
{
    "shell_cmd": "pandoc -o \"$file.pdf\" \"$file\" && open -a Preview \"$file.pdf\"",
    "selector": "text.html.markdown",
    "path": "/usr/texbin:$PATH"
}
```

If you're not on a Mac:

```
{
    "shell_cmd": "pandoc -o \"$file.pdf\" \"$file\"",
    "selector": "text.html.markdown",
    "path": "/usr/texbin:$PATH"
}
```

Save your new custom build to Sublime Text 3/Packages/User and name it as you desire.

Now after saving go back to **Tools > Build System** and select your new build. Now whenever you are writing your markdown and wish to convert it to PDF just press **cmd+b** or **f7** and it will convert it on the spot.

PS: If you wish to edit your custom build, you can find your build at this location:

`~/Library/Application Support/Sublime Text 3/Packages/User/`

Or just use this command: `subl ~/Library/Application Support/Sublime Text 3/Packages/User/YOURBUILDNAME.build`

Custom template (optional)

Grab your custom LaTeX template and edit your custom build that you made in the previous step.

```
{
  "shell_cmd": "pandoc -o \"\$file.pdf\" \"\$file\" && open -a Preview \"\$file.pdf\" --temp
  "selector": "text.html.markdown",
  "path": "/usr/texbin:$PATH"
}
```

Where ‘Niels’ is your Username of your Macbook and ‘default.latex’ is the name of your LaTeX template.

I grabbed my template from:

<https://github.com/jgm/pandoc-templates/blob/master/default.latex>

After the first line starting with `\documentclass`, you can start writing your custom packages / fonts as you desire. Here’s an example of what I added on line 2 to start off with:

```
%%%%%%%% HELVETICA %%%%%%%%%
%\usepackage[scaled=0.86]{helvet}
%\renewcommand\familydefault{\sfdefault}
%\usepackage[T1]{fontenc}
%%%%%%%% OPEN SANS %%%%%%%%%
%\usepackage[default,osfigures,scale=0.95]{opensans}
%\usepackage[T1]{fontenc}
```

Just remove the % from the font you wish to use. For more fonts go to: <http://www.tug.dk/FontCatalogue/sansseriffonts.html>

Example

An example showing how the file looks after converting to PDF is added to the repository with the name `readme.md.pdf`.