

# A Test File to Export MD to PDF with Minted

Ryan

January 8, 2020

## Contents

<b>Contents</b>	<b>1</b>
<b>this is my Index</b>	<b>1</b>
<b>Notes that I would need to map to</b>	<b>2</b>
<b>Useful Commands in vimwiki</b>	<b>2</b>
Tags . . . . .	2
<b>Generated Links</b>	<b>2</b>
List all Note Files . . . . .	4
List all Note Files . . . . .	6
.1     Exporting to PDF . . . . .	7

## Contents

### this is my Index

This [888](#) is just a pain in the ass [888 999](#) aj

[Programs That I've Tried](#)

[how are multi words dealt with Cauchy Example](#)

[blah](#)

OK, so golum is happy with any of the links, but other apps are not, so really the question comes down to whether or not it's work sacrificing MWeb, which *just works* for want of this buton link generation.

\$\$

$$\int_b^a 4x^3 \, dx \tag{1}$$

It's at least a solid start (2)

$$\int_a^b 4x^3 \, dx \tag{3}$$

\$\$ All that needs to be implemented now is a browser of sorts (But I'm gonna do the wiki thing) and I'm set, maybe link to some bash scripts that I can source to export as pdf.

text underneath?

## Notes that I would need to map to

### Useful Commands in vimwiki

- `:VWB` shows a backlinks buffer
- `:VWS` shows a search, but its not fuzz-live-search-with-preview so it's nothing too special
- `:VimwikiGenerateLinks` Insert the links of all available wiki files into the current buffer.
- `:VimwikiTOC`

Create or update the Table of Contents for the current wiki file. See

`vimwiki-toc` .

- `:VimwikiCheckLinks`

Search through all wiki files and check if the targets of all wiki links and links to external files actually exist. Check also if all wiki files are reachable from the index file. The results are shown in the quickfix window.

- `:VimwikiTOC`

Create or update the Table of Contents for the current wiki file. See

`vimwiki-toc` .

- `:VimwikiCheckLinks`

Search through all wiki files and check if the targets of all wiki links and links to external files actually exist. Check also if all wiki files are reachable from the index file. The results are shown in the quickfix window.

### Tags

---

Tags are done like this `:tag1:tag2:`

### Generated Links

- [888](#)
- [999](#)  
anothernote

- [[01 - The Data Directory]]
- [[02 - The Sidebar]]
- [[03 - The Middlebar]]
- [[04 - The Mainbar]]
- [[05 - Notes]]
- [[06 - Tags]]
- [[07 - Shortcuts]]
- [[08 - Importing]]
- [[11 - Synchronization]]
- [[12 - Mobile Editing]]
- [[13 - Collaborative Editing]]
- [[14 - Version Control]]
- [[15 - Encrypted Notes]]
- [[2019-10-30T08-44-28]]
- [[2019-11-01]]
- [[2019-11-02]]
- [[2019-12-22T00-17-47]]
- [[2019-12-22T00-17-47m]]
- [[2019-12-22T00-23-18]]
- [[2019-12-23T01-20-26]]
- [[2019-12-24T23-53-07]]
- [[2019-12-25T00-05-00]]
- [[2019-12-25T00-55-51]]
- [[2019-12-25T01-55-27]]
- [[2019-12-25T04-18-38]]
- [[2019-12-25T04-26-38]]
- [[2019-12-31\_13-51\_delme]]
- [[Abstract Algebra Topic Summary]]
- [[Add Arrow to the end of a line]]
- [[Adding text to a file]]
- [[All Notes in Notable]]
- [[All Notes in Notable (DESKTOP-LKUIQQM's conflicted copy 2019-04-19)]]
- [[Arrays in Java]]
- [[Assigning Solve Solutions in Mathematica]]
- [[Bash Scripting Input]]
- [[Cauchy]]
- [[Chapter 1]]
- [[Charts in Markdown]]
- [[Clear Variables in Mathematica]]
- [[ClipboardConversion\_Notetaking]]
- [[Coding Style]]
- [[Collapsible markdown]]
- [[Combine HTML Files]]
- [[Convert Mathjax to MathML]]
- [[Converting with Pandoc]]
- [[Create LaTeX from Rmd]]
- [[Creating Sets in Java]]
- [[Ctl]]

- [[Cutting off extensions in bash Script]]
- [[Define a Matrix]]
- [[Dictionary Lookup]]
- [[Dictionary Lookup (2)]]

???LINES MISSING - - - -

use ~, i l ~ to insert a link in markdown anyway, its fuzzy down the directory names, first step rename everything to atleast 5 words with underscores

[Notes on Physics](#)

I really reckon that emacs will have a wiki link fixing tool tbh.

## List all Note Files

---

All note files could be listed like so:

```

1  #+begin_src bash
2  # This would work but it's awful
3  $(ls */**/*/*/*/* * *org *md) >> /home/ryan/vimwiki/index.md
4
5  # This is better and it returns `./` which is necessary for /MWeb/ app
6  tldr find
7  find ./ -name \*.org -or -name \*.md -or -name \*.markdown >>
   ↪ /home/ryan/vimwiki/index.md
8
9  # you should put Notebooks / Tags (less than 3) in the file name and
   ↪ then you could use:
10 find ./ -name \*.org -or -name \*.md -or -name \*.markdown | fzf

```

*#+ENDSRC*

which would return:

```

./daily-2020-01-02.md
./Add Arrow to the end of a line.md
./Input Dialogs for Clara.md
./grep.md
./daily-2019-10-30.md
./Charts in Markdown.md
./Physics.md
./Set Working Directory and Load Packages.md
./02 - The Sidebar.md
./index.md
./Chapter 1.md
./01 - The Data Directory.md
./12 - Mobile Editing.md
./2019-12-22T00-23-18.md

```

```
./Untitled (11).md
./Efficiently.md
./Untitled (4).md
./Types of and in Bash.md
./Abstract_Algebra_Org-Mood_Test.org
./03 - The Middlebar.md
...
```

in order to test if this file is referenced anyway, you could reference this list against a me

```
#+begin_src bash :session main
pwd
# Join files togethr
cat $(find ./ -name \*.org -or -name \*.md -or -name \*.markdown) > All.Merged

# Create a list of the files
find ./ -name \*.org -or -name \*.md -or -name \*.markdown > fileList.merged

# Loop for the STDIN of files
# for i in $(find ./ -name \*.org -or -name \*.md -or -name \*.markdown) ; do
#   echo $i
#   done

# Loop for the name of files as read line by line from file
# while read line; do
#   echo "$line"
#   done <fileList.merged

# grep -F "*Alt*" ./KDE.md

# if [grep -F "*Alt*" ./KDE.md]
# then do
#   echo "MATCH!"
# fi
#+end_src
```

```
bash: bind: warning: line editing not enabled bash:
```

```
bind: | warning: | line | editing | not | enabled | | bash: | bind: | warning: | line | editing | not | enabled
| | bash: | bind: | warning: | line | editing | not | enabled | | /home/ryan/vimwiki | | | | |
```

- OK so the search is actually really good, by using preview it is basically live ish
  - I wonder if I could recreate this in emacs somewhat? the issue is all with the preview
- I need to be able to align-right the all links export - - - -

use ~, i l ~ to insert a link in markdown anyway, its fuzzy down the directory names, first step rename everything to atleast 5 words with underscores

## Notes on Physics

I really reckon that emacs will have a wiki link fixing tool tbh.

## List all Note Files

---

All note files could be listed like so:

```
1  #+begin_src bash
2  # This would work but it's awful
3  $(ls */**/*/*/*/* * *org *md) >> /home/ryan/vimwiki/index.md
4
5  # This is better and it returns `./` which is necessary for /MWeb/ app
6  tldr find
7  find ./ -name \*.org -or -name \*.md -or -name \*.markdown >>
   ↪ /home/ryan/vimwiki/index.md
8
9  # you should put Notebooks / Tags (less than 3) in the file name and
   ↪ then you could use:
10 find ./ -name \*.org -or -name \*.md -or -name \*.markdown | fzf
```

*#+ENDSRC*

which would return:

```
./daily-2020-01-02.md
./Add Arrow to the end of a line.md
./Input Dialogs for Clara.md
./grep.md
./daily-2019-10-30.md
./Charts in Markdown.md
./Physics.md
./Set Working Directory and Load Packages.md
./02 - The Sidebar.md
./index.md
./Chapter 1.md
./01 - The Data Directory.md
./12 - Mobile Editing.md
./2019-12-22T00-23-18.md
./Untitled (11).md
./Efficiently.md
./Untitled (4).md
./Types of and in Bash.md
./Abstract_Algebra_Org-Mood _Test.org
./03 - The Middlebar.md
...
```

in order to test if this file is referenced anyway, you could reference this list against a mer

```

#+begin_src bash :session main
pwd
# Join files togethr
cat $(find ./ -name \*.org -or -name \*.md -or -name \*.markdown) > All.Merged

# Create a list of the files
find ./ -name \*.org -or -name \*.md -or -name \*.markdown > fileList.merged

# Loop for the STDIN of files
# for i in $(find ./ -name \*.org -or -name \*.md -or -name \*.markdown) ; do
#   echo $i
# done

# Loop for the name of files as read line by line from file
# while read line; do
#   echo "$line"
# done <fileList.merged

# grep -F "*Alt*" ./KDE.md

# if [grep -F "*Alt*" ./KDE.md]
# then do
#   echo "MATCH!"
# fi
#+end_src

```

bash: bind: warning: line editing not enabled bash:

```

bind: | warning: | line | editing | not | enabled | | bash: | bind: | warning: | line | editing | not | enabled
| | bash: | bind: | warning: | line | editing | not | enabled | | /home/ryan/vimwiki | | | | |

```

- OK so the search is actually really good, by using preview it is basically live ish
  - I wonder if I could recreate this in emacs somewhat? the issue is all with the preview
- I need to be able to align-right the all links export
- I need to be able to see which of those is orphaned
- Creating files automagically from vimwiki isn't really a feature, I could already do that from emacs in a just as good, if not better way

## Exporting to PDF

In order to export to PDF, just use pandoc with a .sty header, however, when using the minted package (which is preferred for the broader language support and nice formats) it is necessary to pass the shell-escape flag.

Also pandoc doesn't all align to occur inside  $$$$  like mathjax does, so you'll need to use something like `:s/{align}/{aligned}/g` and to get rid of labels maybe `:s/\label{[\w\W]*}//g`, then just execute the following in the terminal:

```
1  # Use shell-escape because minted is loaded as a package
2  # Pandoc won't use it but it throws an error otherwise
3  pandoc -s --self-contained index.md --toc -H
   ↪ /home/ryan/Dropbox/profiles/Templates/LaTeX/ScreenStyle.sty
   ↪ --pdf-engine-opt=-shell-escape -o index.pdf
```

or if you wanted to use the minted package you would have to do it manually, it's not supported by pandoc:

```
1  pandoc -s index.md --toc --listings -H
   ↪ /home/ryan/Dropbox/profiles/Templates/LaTeX/ScreenStyle.sty
   ↪ --minted -o index.tex
2  # This sed won't Work!
3  sed -i 's/\lstlisting/minted/g' index.tex
4  xelatex -shell-escape -synctex=1 -interaction nonstopmode
   ↪ -output-directory ./ index.tex
```

Actually it might be easier to go from md to org and then just use the org export pathway you've already got set up

This was converted from 'md' to 'org' using 'pandoc -f gfm' at time: 2020-01-07T22-23-06