Saluations, and welcome to this beginner’s guide to creating and converting academic texts using pandoc and its feature-packed flavour of Markdown! On a basic level, you can simply read through the tutorial to see what Markdown consists of and how it works. To use the language yourself, however, you will need:

* a text editor – to view and edit Markdown source files. Any editor, from Windows Notepad on up, will suffice, though more advanced programs such as [Sublime Text](https://www.sublimetext.com/) provide a more enjoyable writing environment.
* pandoc – both a download link and installation instructions are available on John MacFarlane’s [pandoc website](https://pandoc.org/) (link at the end of the Markdown post).
* As the website explains, pandoc is a command-line utility; that is, it is operated entirely by typing in commands at the Windows Shell or Linux/OS X terminal. This may at first give you pause, but as we will see, very little typing is actually required, and the basics of pandoc are very quick to get to grips with.

I should also probably say a word or two on file formats. Markdown files are commonly saved with an .md, .mdown or even .markdown extension, and you should be able to use any of these interchangably. If you’re using pandoc, however, they can also simply be saved as normal plain text files (.txt), and that’s the way I prefer to do things. I like the idea of using a more universal file format – this way, even if I switch to a text editor that doesn’t recognize .md files (an older one, for instance), or Markdown becomes less popular, I’ll at least be able to read and edit my work. If you decide to switch from pandoc to another Markdown editor (there are a bucketload), it would be worth checking which extensions it supports, but otherwise the choice is entirely aesthetic.

With that preamble dealt with, on with the tutorial…

# Markdown and pandoc mini-tutorial

This is the first paragraph of our great opus. Paragraphing in Markdown is achieved simply by leaving a blank line in between two blocks of text. Like this…

This is now paragraph 2. Bear in mind that these paragraphs are *semantic* units of text, not presentational; they will not necessarily appear like this in whichever format we convert this file to, but will be affected by the paragraphing style of whichever program we use to view it. To see an example of this, compare these paragraphs in the .txt source file with those in the converted .docx file. That’s enough text for now, so we’ll move on the first sub-section of this piece.

## Sub-section 1

That sub-heading above is also a semantic unit, and will also be formatted appropriately for each file type. In some cases, it will also be used in the organization of the converted document; Word and LibreOffice, for instance, use headings as markers for their ‘navigation’ panes. If the converted file type allows for it, pandoc can also use the headings and sub-headings of a document to generate a table of contents (though you’ll have to read the documentation to find out how that’s done). Sub-headings can be nested if desired, like so:

### Nested sub-section

Some text here…

#### Even more nested sub-section

And so on, up to six levels deep. Markdown also recognizes headings produced using equals signs (=) and hyphens (-) to underline the text. The choice between these methods is purely aesthetic, and will not affect the appearance of headings in the converted file—though consistency within a single document is preferable to avoid confusion.

## Sub-section 2: Eye candy

Text can be emphasized in two ways, both of which rely on asterisks. *Single asterisks* normally create italicized text, while **double asterisks** make text bold. Underscores ( \_ ), either singly or doubly, can be used as an alternative. en- and em-dashes, on the rare occasions when they are required, can be produced using a set of two (–) or three (—) hyphens, respectively.

Lists can also be created with ease. ‘Unordered’ lists, which usually translate to bulleted lists in the converted file, are marked with asterisks, pluses (+) or hyphens, interchangably:

* list item
* a second list item
* a third list item
	+ a nested list item
	+ and so on…

‘Ordered’ lists, on the other hand, use numbers followed by full stops:

1. list item
2. another list item
3. this is, presumably, clear by now

Though the actual numbers used in an ordered list after the first number 1 can be arbitrary (again, they will be replaced in the converted file with appropriate numbering settings), for clarity’s sake it is best to number items consecutively.

List items may contain multiple paragraphs, if the first line of each paragraph is indented (by at least four spaces or one tab):

* This is a list item.
* This is the second paragraph of the same list item.
* This is another list item.

Finally, we’ll add a block quote to this section. For these, Markdown takes inspiration from plain text email, and uses a single angle bracket (>) at the beginning of the quote:

What is a man
If his chief good and market of his time
Be but to sleep and feed? A beast, no more.
Sure he that made us with such large discourse,
Looking before and after, gave us not
That capability and godlike reason
To fust in us unus’d.
– *Hamlet, IV. 4. 33-39 (Arden Shakespeare edition, ed. Harold Jenkins, 1982, repr. 2005)*

Though it may be difficult to see here, it’s important to note that each line ends with a double-space; this is to instruct pandoc to start a new line.

Naturally, there are various additional formatting options, such as horizontal rules, tables, hyperlinks and so on, but the above represents the majority of what most writers in the humanities are ever likely to use regularly.

## Sub-section 3: References

While pandoc’s version of Markdown includes a plugin for handling citations, I’ve always found it rather unreliable (particularly where the formatting of the resulting references is concerned), and I’m not going to go into it here.[[1]](#footnote-29) It’s far more manageable, at least for footnote- or endnote-based citation styles, to use Markdown’s own footnote system.

Footnotes are inserted by placing a placing a footnote reference, preceded by a circumflex character (^), within square brackets[[2]](#footnote-30). The footnote itself can be included anywhere in the document, provided the reference matches that included in the text. The reference can be any combination of words and letters, though for longer, more densely referenced documents, you are advised to develop a consistent system that makes sense to you. It may even be advisable in such cases to keep all footnote references in a separate Markdown file—as we’ll see shortly, pandoc can combine multiple files into a single output document.

If your preferred citation style calls for endnotes instead, the footnotes should be switched after the file has been converted (as has been done in the .docx version of this document). In Word, for instance, this can be done from a dialog box in the ‘References’ tab.[[3]](#footnote-31)

The contents of the footnote (or rather, the way in which you insert your actual citations) will depend on your bibliographic software and preferred working method. The most basic method is to copy and paste each citation as plain text, as in this example.[[4]](#footnote-32)

This has the disadvantage of being disconnected from your bibliographic database, and citations produced in this way will not be automatically updated if you edit that database. Nor can references be adapted for different citation styles, save by recopying or retyping each one. Unfortunately, this is one area in which plain text writing lags behind word processing; there’s no equivalent of the plugins offered by Word and LibreOffice for popular reference managers such as EndNote or Zotero. For Zotero users, however, this difficulty can be overcome with a modicum of investigation and willingness to experiment, as well as a third-party plugin.[[5]](#footnote-33)

## Sub-section 4: Conversion

The completed Markdown document can be converted from the command line with a single pandoc command. Open the command-line program appropriate to your operating system (Power Shell or Command Prompt for Windows or some variant on the word ‘terminal’ for Mac or Linux users), and navigate to the folder containing your Markdown source file. Type:

pandoc -S -o output-file.docx input-file.txt

To take the elements of this command in order:

1. The ‘-S’ option instructs pandoc to use ‘smart’ formatting when converting the document: changing straight quotation marks into curly quotes, a series of full stops into an elipsis, and a series of hyphens into an en- or em-dash, as detailed above.
2. The ‘-o’ option tells pandoc to output the conversion to a file (if this is not included, pandoc will simply display the converted document on screen without saving it, which is of limited use).
3. ‘output-file’ refers to the file in which pandoc will save the converted document. If this file does not exist, it will be created.
* The file extension (here, ‘.docx’) will inform pandoc of the file type for the conversion. To convert the Markdown document to a web page instead, for instance, we would type: ‘output-file.html’.
1. Unsurprisingly, ‘input-file’ refers to the document to be converted.

Pandoc is also capable of combining a series of input files into a single output file:

pandoc -S -o output-file.docx input-file1.txt input-file2.txt citations.txt

By default, all files are appended in the order listed to the first input file. Here, for example, the contents of ‘input-file2’ would be added to the end of ‘input-file1’. If the third file, ‘citations’, lives up to its name and contains only footnote references, it can be included anywhere in the input list, and the footnotes will be added in the appropriate locations.

## Sub-section 5: Reference files (.docx or .odt)

When converting files in this way, pandoc will apply a set of default formatting options for the output file type; the resulting file will then require considerable editing to conform to the standards set out by your university, the journal or publisher you’re submitting to, or even your personal preferences. If these standards are largely the same for each document you write, however, substantial time can be saved by using a *reference file*. This is simply a file of the appropriate output type (currently, this is available only for Word or LibreOffice files) whose formatting has been edited as desired.[[6]](#footnote-36) The reference file is then included in the conversion command:

pandoc -S -o output-file.docx –reference-docx=reference-file.docx input-file.txt

or…

pandoc -S -o output-file.odt –reference-odt=reference-file.odt input-file.txt

Note the double-hyphen in the filter including the file (this will, of course be visible only in the Markdown file, as the hyphens will be converted to an en-dash in the .docx version). Pandoc will then use the formatting instructions (style sheets, in the case of Word files) of the reference file when generating the new output file.

The principal advantage of this, of course, is that the reference file needs to be created and edited only once, and can then be used to guide the formatting of as many documents as desired. Conceivably, different reference files could be produced for different journals, for book as opposed to journal publication, or different types of coursework.

## Conclusion

And that, sadly, brings this epic to a close. As mentioned earlier, the most effective way to gauge the power and flexibility of the Markdown-pandoc partnership, as well as to determine whether it is suitable for your own needs, is to try it. Comparing this Markdown document with its output would be a useful start. You could then install pandoc and convert this file to other formats, noting what does and doesn’t change. Hopefully, the result will be an improved workflow that offers you a more focused connection with your academic writing, in a more lightweight, mobile and stable environment.

1. For more information on this, see pandoc’s reference guide: John MacFarlane, *Extension: citations* [n.d.] <http://johnmacfarlane.net/pandoc/demo/example19/Extension-citations.html> [accessed 15 December 2014] [↑](#footnote-ref-29)
2. Here we are, then! [↑](#footnote-ref-30)
3. Microsoft Corporation, *Convert footnotes to endnotes and vice versa* (2014) <http://office.microsoft.com/en-gb/word-help/convert-footnotes-to-endnotes-and-vice-versa-HP001231815.aspx> [accessed 15 December 2014] [↑](#footnote-ref-31)
4. Robert Service, *Comrades: A World History of Communism* (London: Macmillan, 2007), pp. 25-31. [↑](#footnote-ref-32)
5. That investigation could do worse than start here: The Zoteroist, *Announcing RTF/ODF Scan for Zotero* (2013) <http://zoteromusings.wordpress.com/2013/05/06/announcing-rtfodf-scan-for-zotero/> [accessed 15 December 2014]. The instructions refer to Scrivener and Google Docs, but are equally applicable to plain text writing. For more details on using this plugin and integrating it into your workflow, please do get in touch; we are more than aware that efficient referencing is at once the most important and the most convoluted aspect of this writing method. [↑](#footnote-ref-33)
6. For more details, see: The Hackademic, *Pandoc and Academic .docx files* (2014) <http://hackademic.postach.io/pandoc-and-academic-docx-files> [accessed 15 December 2014] [↑](#footnote-ref-36)