

Academic Markdown

Who knew writing could be so nerdy?

Copyright 2013 Sebastiaan Mathôt

Contents

- [About](#)
- [Basic usage](#)
- [Dependencies](#)
- [Zotero references](#)
- [Academic Markdown extensions](#)
 - [figure: figures](#)
 - [exec: external commands](#)
 - [include: include other Markdown files](#)
 - [toc: table of contents](#)
- [Styling](#)
- [License](#)

About

Academic Markdown is a Python module that allows you generate .pdf, .docx, and .odt files from Markdown source. [Pandoc](#) is used for most of the heavy lifting, so please refer to the Pandoc website for detailed information about writing in Pandoc Markdown. However, Academic Markdown offers some additional functionality that is useful for writing scientific documents, such as integration with the [Zotero](#) reference manager.

Basic usage

```
from academicmarkdown import build
build.HTML('input.md', 'output.html')
build.PDF('input.md', 'output.pdf')
build.DOCX('input.md', 'output.docx')
build.ODT('input.md', 'output.odt')
```

Dependencies

Academic Markdown has been tested only on Ubuntu Linux. The following dependencies are required:

- `pandoc` is used for most of the heavy lifting.
- `pyzotero` is necessary for extracting Zotero references.
- `wkhtmltopdf` is necessary for converting to .pdf.

Zotero references

Since the basic Markdown conversion is performed by Pandoc, citations should be formatted as described on the Pandoc site:

- <http://johnmacfarlane.net/pandoc/README.html#citations>

You can automatically extract references from your Zotero library by setting the `zoteroApiKey` and `zoteroLibraryId` properties. You can find your API key and library ID online on your Zotero profile page.

```
from academicmarkdown import build
build.zoteroApiKey = u'myapikey'
build.zoteroLibraryId = u'mylibraryid'
build.PDF(u'input.md', u'output.pdf')
```

Once you do this, citations will be matched using the following logic. Let's consider the citation `[@MathSchrThee2012OpensesameBehav]`:

1. The citation is split into separate words based on capitalization. So in this case: `math + schr + thee + 2012 + opensesame + behav`.
2. All words before the year are matched to author names.
3. The year is matched to the year of publication.
4. All words after the year are matched to titles of the paper and the journal/ book.
5. If the citation matches multiple entries in your Zotero database, one will be chosen at random.

Previous references will be cached automatically. To refresh, remove the file `.zoteromarkdown.cache` or run your Python script with the command-line argument: `--clear-cache`.

Academic Markdown extensions

Academic Markdown provides certain extensions to regular Markdown, in the form of YAML blocks embedded in `%-- --%` tags.

figure: figures

The `figure` block embeds a Figure in the text. Figures are numbered automatically. The ID can be used to refer to the Figure in the text, using a `%` character. So the following figure would be referred to as `%FigFA`.

```
%--
figure:
  id: FigFA
  source: example/foveal_acuity.png
  caption: "Visual acuity drops of rapidly with distance from the fovea"
--%
```

exec: external commands

The `exec` block inserts the return value of an external command in the text. For example, the following block embeds something like ‘Generated on 10/18/2013’:

```
%-- exec: "date +'Generated on %x'" --%
```

include: include other Markdown files

The `include` block includes an other Markdown file. For example:

```
%-- include: example/intro.md --%
```

toc: table of contents

The `toc` block will automatically generate a table of contents from the headings, assuming that headings are indicated using the `#` style and not the underlining style. You can indicate headings to be excluded from the table of contents as well.

```
%--
toc:
  mindepth: 1
  maxdepth: 2
  exclude: [Contents, Contact]
--%
```

Styling

You can specify a style by setting the `style` attribute:

```
from academicmarkdown import build
build.style = u'my-style'
build.PDF(u'input.md', u'output.pdf')
```

The `style` attribute must be a folder that contains a number of files (not all files need to exist):

- `html5.html` is the HTML5 template used for generating HTML and PDF documents.
- `html5.css` is the stylesheet template used for generating HTML and PDF documents.
- `reference.csl` is a CSL citation style.
- `wkhtmltopdf.tpl` is a shell command that is used to convert HTML to PDF. It can have something like the following contents:
`wkhtmltopdf -L 20 -R 20 "%(source)s" "%(target)s"`

License

Academic Markdown is available under the GNU General Public License 3. For more information, see the included file `COPYING`.