Another markdown to pdf tool: md2pdf

Another markdown to pdf project. This time including indexed toc generation.

Requirements

wkhtmltopdf

Install from: https://wkhtmltopdf.org/ .

popper’s pdftohtml

Unix-like systems install popper/libpopper utils. Windows: install from https://blog.alivate.com.au/poppler-windows/

Nodejs

You must have installed or installa `nodejs` for your platform with the following modules (with the following minimal version) :

- markdown-it: 12.0.2
- markdown-it-anchor: 6.0.0
- markdown-it-table-of-contents: 0.5.1
- strip-bom: 4.0.0

Usage

```
python -m amd2pdf [-v] [-d] [-O] [-c/--css filename.css] [-p/--page PAGE-TYPE]
[-t/--title title] [-o/--output filename] input_filename.md
 -v: set verbose mode on
 -d: set debug mode on
 -O: open output in browser
 -t title: set document title
 -o filename: set output filename
 -c/--css filename.css: stylesheet to use, default: style.css
 -p/--page PAGE-TYPE: Default A4

Output will be: input_filename.pdf
```

Environment options

Headers and footers

You can set environment variables:

`[WHERE]_[POSITION]="something important"`

where:

- WHERE is `HEADER` or `FOOTER`
- POSITION is `LEFT` , `CENTER` or `RIGHT`

Examples:

- `set HEADER_LEFT="This is my document"`
- `set FOOTER_CENTER="[page] of [topage]"`

Special variables

You can use some special variables in the headers and footers:

- `[page]` Replaced by the number of the pages currently being printed
- `[frompage]` Replaced by the number of the first page to be printed
- `[topage]` Replaced by the number of the last page to be printed
- `[webpage]` Replaced by the URL of the page being printed
- `[section]` Replaced by the name of the current section
- `[subsection]` Replaced by the name of the current subsection
- `[date]` Replaced by the current date in system local format
- `[isodate]` Replaced by the current date in ISO 8601 extended format
- `[time]` Replaced by the current time in system local format
- `[title]` Replaced by the title of the of the current page object
- `[doctitle]` Replaced by the title of the output document
- `[sitepage]` Replaced by the number of the page in the current site being converted
- `[sitepages]` Replaced by the number of pages in the current site being converted

More detail about this in wkhtmltopdf documentation.

Other Parameters

- `TITLE` : document's title
- `HEAD` : head section to include in intermediate html
- `PAGE` : the page size

TOC levels

Toc levels to include:

- `TOC_INCLUDE_LEVEL` : json encoded levels array. Default is `[2, 3]` which makes ToC generator to include 2 and 3 headings in toc.

Toc tag used in markdown is `[[TOC]]` .

Examples

Some exemplary text

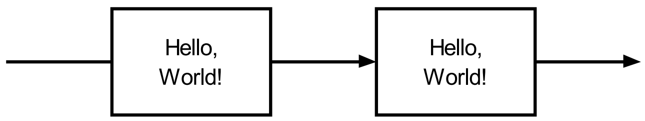
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

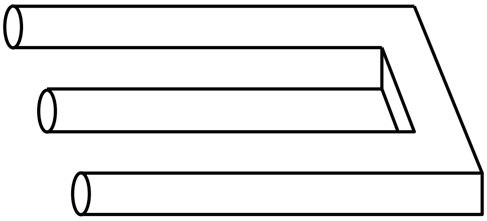
> At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae.
>
> > Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat

```
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur ac mattis diam, sit amet fermentum ex. Cras elementum at ur
na ut interdum. Nullam at felis risus. Ut tortor turpis, porttitor et consectetur vel, varius eget purus. Sed dolor ipsum, vulp
utate nec est eget, dignissim sagittis metus. Nullam ut velit elit. Phasellus sit amet nisi eu sem sagittis vestibulum.
```

Pikchr diagrams!



or:

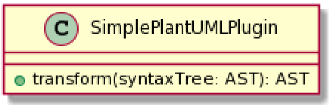


Syntax highlighting

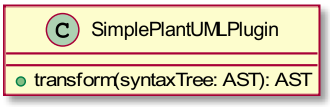
```
def main():
    print("something")
```

Plantuml

standard image output



vectorial output



text output

```
,---------------------------------.
|SimplePlantUMLPlugin             |
|---------------------------------|
|+ transform(syntaxTree: AST): AST|
`---------------------------------'
```

This same readme converted to pdf:

- README.sample1.pdf
- README.sample2.pdf