

1 `md2tex` - a CLI to convert Markdown to TeX

`md2tex` is a simple but customizable markdown to TeX converter. rather than using fancy object oriented libraries, it locates elements of markdown syntax using regular expressions and translates them into TeX syntax. it has only been tested with latin script so far. still a work in progress.

1.1 Features

- translation from markdown to TeX
 - creation of a complete and valid TeX document, either from a generic template or using your own template
 - saving to a custom file(path)
 - several options for extra customization
-

1.2 Markdown syntax

currently supported

- line breaks: paragraph changes using `\n\n`, `<br` and `<br/`
- **bold**, *italic* and `inline code` text. **warning**:
 - only bold and italics made using `*` will be translated
- block quotes (lines beginning with `)`. **warning**: only non-nested block quotes – or the outer level of a nested block quote – will be rendered.
- multiline code; if possible, the code is colored thanks to `minted`
- ordered and unordered lists, including nested lists. **warning** :
 - this CLI is made to work with as-messy-as-possible data; if your lists are messy (using different levels of space...), it will create messy nested lists too.
 - unordered lists will only be converted if they begin with `-`. any other list token will be ignored.
- titles and different title levels
- images

- URLs
- inline quotes : french and english style, nested quotes.
- footnotes ; **warning**: loose footnotes (references in the body that point to nothing or footnotes that point to nothing in the body) will be deleted.

unsupported

- markdown tables
- end-of-line backslashes to mark a line break
- strikethrough
- emojis
- tasks lists
- definition lists
- text blocks inside lists. they will be processed as the continuation of a list item

1.3 Things that will mess with the output

- poorly formed markdown
- things that are currently not supported
- unnumbered lists made using non-standard markdown syntax (lists not beginning with -...)
- numbered lists using additional numbering keys, such as 1. a. in this case, 1. will be deleted and a will be kept, so the TeX file will contain double numbering (the `enumerate` numbering and this numbering key)
- highly nested lists are not supported by default in LaTeX, but this can be changed in your preamble
- multiple footnotes with the same key; for the footnote restructuration to work, there must be unique footnote numbers and only one note per footnote number.
- loose footnotes that point to nothing will be deleted

clean your file first ! you can also make the bulk of the translation using this cli and fine tune your `.tex` file later on.

1.4 About nested lists and visual indentation

list nesting and visual indentation is a bit of a pickle, so here's an explanation of how the indentation is processed and what constitutes a valid list for this tool.

what is a valid list?

- all items must be **as indented than the first list item** or more. the indentation level of the first item will be removed before processing that list. this means that the first example will result in the same result as the second

```
- list item with no leading space
- same here
```

```
- list item with two leading spaces
- same here, still a valid list
```

- all indentation levels must be a multiple of the leading spaces of the first nested item. if the first nested list item is indented with 4 whitespaces, all nested items must be indented with a multiple of 4 spaces. for example, a list item can't be indented with 3 spaces while another one is indented with 4 spaces.
- the above two rules cause the script to exit with an error message pointing to the faulty list. on top of that, in keeping with LaTeX and markdown logic, a list item can't skip a nesting level. this will **not** stop the script, the indentation will just be reset automatically.

```
- this is an item at level 0
  - this is an item at level 1; the indentation is set as 2 whitespaces
  - same here; level 1.
    - visually, this one seems indented as level 3 as it starts with 6
    ↪ leading whitespaces.
    however, its nesting level will be reset from level 3 to level 2 to
    ↪ avoid skipping
    a nesting level
  - this item is at level 1
```

technically, how does it work ?

- the complete markdown list is located

- each list item is mapped to its indentation level, in number of leading spaces.
 - the list is validated. if it is not valid, the script stops.
 - an indentation multiplier is defined: it is the number of spaces in the first nested list item
 - this multiplier is used to map each list item to its nesting level
 - the markdown is syntax is replaced my markdown syntax.
-

1.5 License

released under GNU GPL v3.

```
md2tex README.md -c -t ./readme/readme_article_template.tex -o ./readme/README_article  
-d article -u
```

Contents