

NAME

mkd_functions — access and process Markdown documents.

LIBRARY

Markdown (libmarkdown, -lmarkdown)

SYNOPSIS

```
#include <mkdio.h>

int
mkd_compile(MMIOT *document, int flags);

int
mkd_css(MMIOT *document, char **doc);

int
mkd_generatecss(MMIOT *document, FILE *output);

int
mkd_document(MMIOT *document, char **doc);

int
mkd_generatehtml(MMIOT *document, FILE *output);

int
mkd_xhtmlpage(MMIOT *document, int flags, FILE *output);

int
mkd_toc(MMIOT *document, char **doc);

void
mkd_generatetoc(MMIOT *document, FILE *output);

void
mkd_cleanup(MMIOT*);

char*
mkd_doc_title(MMIOT*);

char*
mkd_doc_author(MMIOT*);

char*
mkd_doc_date(MMIOT*);
```

DESCRIPTION

The **markdown** format supported in this implementation includes Pandoc-style header and inline *<style>* blocks, and the standard **markdown(3)** functions do not provide access to the data provided by either of those extensions. These functions give you access to that data, plus they provide a finer-grained way of converting *Markdown* documents into HTML.

Given a *MMIOT** generated by **mkd_in()** or **mkd_string()**, **mkd_compile()** compiles the document into *<style>*, *Pandoc*, and *html* sections.

Once compiled, the document can be examined and written by the **mkd_css()**, **mkd_document()**, **mkd_generatecss()**, **mkd_generatehtml()**, **mkd_generatetoc()**, **mkd_toc()**, **mkd_xhtmlpage()**, **mkd_doc_title()**, **mkd_doc_author()**, and **mkd_doc_date()** functions.

mkd_css() allocates a string and populates it with any *<style>* sections provided in the document, **mkd_generatecss()** writes any *<style>* sections to the output, **mkd_document()** points *text* to the text of the document and returns the size of the document, **mkd_generatehtml()** writes the rest of the document to the output, and **mkd_doc_title()**, **mkd_doc_author()**, **mkd_doc_date()** are used to read the contents of a Pandoc header, if any.



mkd_xhtmlpage() writes a xhtml page containing the document. The regular set of flags can be passed.

mkd_toc() writes a document outline, in the form of a collection of nested lists with links to each header in the document, into a string allocated with **malloc()**, and returns the size.

mkd_generatetoc() is like **mkd_toc()**, except that it writes the document outline to the given **FILE*** argument.

mkd_cleanup() deletes a **MMIOT*** after processing is done.

mkd_compile() accepts the same flags that **markdown()** and **mkd_string()** do;

MKD_NOIMAGE	Do not process '!' and remove <code></code> tags from the output.
MKD_NOLINKS	Do not process '[' and remove <code><a></code> tags from the output.
MKD_NOPANTS	Do not do Smarty-pants-style mangling of quotes, dashes, or ellipses.
MKD_TAGTEXT	Process the input as if you were inside a html tag. This means that no html tags will be generated, and mkd_compile() will attempt to escape anything that might terribly confuse a web browser.
MKD_NO_EXT	Do not process any markdown pseudo-protocols when handing <code>[[[</code> links.
MKD_NOHEADER	Do not attempt to parse any Pandoc-style headers.
MKD_TOC	Label all headers for use with the mkd_generatetoc() function.
MKD_1_COMPAT	MarkdownTest_1.0 compatibility flag; trim trailing spaces from the first line of code blocks and disable implicit reference links.
MKD_NOSTRIKETHROUGH	Disable strikethrough support.

RETURN VALUES

The function **mkd_compile()** returns 1 in the case of success, or 0 if the document is already compiled. The function **mkd_generatetoc()** returns the number of bytes written in the case of success, or EOF if an error occurred. The function **mkd_generatehtml()** returns 0 on success, -1 on failure.

SEE ALSO

markdown(1), **markdown(3)**, **mkd-line(3)**, **markdown(7)**, **mkd-extensions(7)**, **mmap(2)**.

<http://daringfireball.net/projects/markdown/syntax>

BUGS

Error handling is minimal at best.

