Package 'md4r'

February 5, 2024

Title Markdown Parser Implemented using the 'MD4C' Library

Version 0.5.2.0

Description Provides an R wrapper for the 'MD4C' (Markdown for 'C') library.
Functions exist for parsing markdown ('CommonMark' compliant) along with support for other common
markdown extensions (e.g. GitHub flavored markdown, 'LaTeX' equation support, etc.). The package also provides a number of higher level functions for exploring and manipulating markdown
abstract syntax trees as well as translating and displaying the documents.

License MIT + file LICENSE

Copyright John MacFarlane, RStudio, PBC; Martin Mitáš, Colin Rundel

Encoding UTF-8

RoxygenNote 7.3.1

LinkingTo Rcpp

Imports tibble, Rcpp, cli, checkmate, glue, purrr, textutils, stringr

Config/testthat/edition 3

Suggests withr, testthat (>= 3.0.0), diffmatchpatch

NeedsCompilation yes

Author Colin Rundel [aut, cre],
Martin Mitáš [cph] (md4c author: md4c.c, md4c.h, specs/md4c/),
RStudio, PBC [cph] (httpuv_url_tools.cpp),
John MacFarlane [cph] (specs/gfm/spec.txt, specs/md4c/spec.txt)

Maintainer Colin Rundel <rundel@gmail.com>

Repository CRAN

Date/Publication 2024-02-05 20:40:06 UTC

R topics documented:

flags *Markdown parser flags*

Description

The md4c library supports a number of markdown variants / options. The parsing of these is controlled by flags passed to the parser. The following functions provide commonly used utilities for these flags.

Usage

```
flags_available()

flags_describe()

flags_used(md)
```

Arguments

md	Markdown ast object

Value

`flags_available()` returns a character vector of available flags accepted by `parse_md()`.

`flags_describe()` returns a tibble with columns `flag` and `description` describing each flag.

`flags_used()` returns a character vector of flags used in a parsed markdown document.

Examples

```

flags_available()

flags_describe()

md_file = system.file("examples/commonmark.md", package = "md4r")
md = parse_md(md_file)
flags_used(md)
```

parse_md *Parse markdown*

Description

Parse either a literal markdown string or a markdown file given a path. Different dialects and features are supported via the `flags` argument. See `flags_describe()` for possible flags and their usage. `parse_md()` defaults parsing using the commonmark spec while `parse_gfm()` uses the GitHub flavored markdown spec.

Usage

```
parse_md(md, flags = "MD_DIALECT_COMMONMARK")

parse_gfm(md, flags = "MD_DIALECT_GITHUB")
```

Arguments

`md`	Character. Either literal string of markdown or a path to a markdown file.
`flags`	Character vector. Dialect flags used by the parser.

Value

Both functions return a markdown ast, a list with the `md_block_doc` class.

Examples

```
parse_md(system.file("examples/commonmark.md", package = "md4r"))

parse_gfm(system.file("examples/github.md", package = "md4r"))
```

to_html *Convert to html*

Description

Coverts a markdown object (full ast or node) to HTML.

Usage

```
to_html(md, ...)
```

Arguments

`md`	Markdown object
`...`	Unused, for extensibility.

Value

Returns a character vector of HTML lines representing the markdown object.

Examples

```
md_file = system.file("examples/commonmark.md", package = "md4r")
md = parse_md(md_file)
cat(to_html(md), sep="\n")
```

`to_md` *Convert to markdown*

Description

Coverts a markdown object (full ast or node) to markdown text.

Usage

```
to_md(md, ...)
```

Arguments

`md`	Markdown object
`...`	Unused, for extensibility.

Value

Returns a character vector of markdown lines representing the markdown object.

Examples

```
md_file = system.file("examples/commonmark.md", package = "md4r")
md = parse_md(md_file)
cat(to_md(md), sep="\n")
```

Index