

Package ‘rdoc’

October 14, 2022

Type Package

Title Colourised R Documentation

Version 0.1.0

Maintainer Matthew de Queljoe <matthew.dequeljoe@gmail.com>

Description

Extends tools::Rd2txt() by adding customisable text and colour formatting to R documentation contents. If used from a terminal, output will be displayed via file.show() otherwise contents will be printed in sections. Also provides stand-in replacements for ?() and help().

URL <https://github.com/mdequeljoe/rdoc>

BugReports <https://github.com/mdequeljoe/rdoc/issues>

License GPL-3

Encoding UTF-8

LazyData true

Imports R6, crayon, cli, prettycode, tools, utils

RoxygenNote 6.1.1

Suggests testthat, mockery, withr

NeedsCompilation no

Author Matthew de Queljoe [aut, cre]

Repository CRAN

Date/Publication 2019-05-16 11:10:03 UTC

R topics documented:

rdoc	2
rdoc_options	3
rdoc_question	3
rdoc_rd	4
rdoc_style	5
rdoc_text	5
rdoc_text_formats	6
use_rdoc	7

`rdoc`*Colourised R documentation*

Description

Refer to colourised R docs as terminal output. Provides a replacement for help. A number of common (substantive) section accessors are also provided.

Usage

```
rdoc(topic, package = NULL, lib.loc = NULL)
```

```
rdoc_usage(topic, package = NULL, lib.loc = NULL)
```

```
rdoc_arguments(topic, package = NULL, lib.loc = NULL)
```

```
rdoc_details(topic, package = NULL, lib.loc = NULL)
```

```
rdoc_examples(topic, package = NULL, lib.loc = NULL)
```

Arguments

`topic` character(1), help topic

`package` character(1), package of help topic. Defaults to NULL

`lib.loc` character(1), library location of package. Defaults to NULL

Details

The manner in which content is outputted depends on where it is being used. If used from a terminal the contents will be outputted via `file.show`. Otherwise the output will be printed interactively by sections unless this has been disabled via `options(rdoc.by_section = FALSE)`. When printing by section pressing the `<enter>` keypress will show the next section. Any other keypress will exit the interaction without printing any remaining sections. Color support is checked via `crayon::has_color()`. Customising rdoc output is possible via ‘options’, see [rdoc_options](#) for more details.

Examples

```
library(rdoc)
rdoc("rdoc")
rdoc_examples("min")
rdoc_usage(substr)
```

rdoc_options	<i>rdoc_options</i>
--------------	---------------------

Description

Rdoc uses the following `options()` to set output:

- ‘`rdoc.by_section`’: logical, output interactively printed by sections for console sessions. Sections will be printed with the <enter> keypress. Any other keypress will exit the interaction without printing the remaining sections. Default is TRUE.
- ‘`rdoc.header`’: logical, whether package header is included in the output. Default is TRUE.
- ‘`rdoc.text_formats`’: see [rdoc_text_formats](#)
- ‘`rdoc.style`’: see [rdoc_style](#)
- ‘`rdoc.item_bullet`’: Symbol to use for itemized lists. Default is `paste0(cli::symbol$circle, " ")`

rdoc_question	<i>Colourised R documentation</i>
---------------	-----------------------------------

Description

Refer to colourised R docs as terminal output. Provides a replacement for `?`. Meant primarily for use as an override in `use_rdoc()`

Usage

```
rdoc_question(type, topic)
```

Arguments

<code>type</code>	the special type of documentation to use for this topic; for example, if the type is <code>class</code> , documentation is provided for the class with name <code>topic</code> . See the Section ‘S4 Method Documentation’ for the uses of <code>type</code> to get help on formal methods, including methods? <i>function</i> and method? <i>call</i> .
<code>topic</code>	Usually, a name or character string specifying the topic for which help is sought. Alternatively, a function call to ask for documentation on a corresponding S4 method: see the section on S4 method documentation. The calls <code>pkg::topic</code> and <code>pkg:::topic</code> are treated specially, and look for help on <code>topic</code> in package <code>pkg</code> .

Details

The manner in which content is outputted depends on where it is being used. If used from a terminal the contents will be outputted via `file.show`. Otherwise the output will be printed interactively by sections unless this has been disabled via `options(rdoc.by_section = FALSE)`. When printing by section pressing the `<enter>` keypress will show the next section. Any other keypress will exit the interaction without printing any remaining sections. Color support is checked via `crayon::has_color()`. Customising rdoc output is possible via ‘options’, see [rdoc_options](#) for more details.

Examples

```
rdoc::rdoc_question('lapply')
```

rdoc_rd

Colourised R documentation

Description

Refer directly to `.Rd` file contents as terminal output.

Usage

```
rdoc_rd(path)
```

Arguments

path character(1), the path to an `.Rd` file

Details

The manner in which content is outputted depends on where it is being used. If used from a terminal the contents will be outputted via `file.show`. Otherwise the output will be printed interactively by sections unless this has been disabled via `options(rdoc.by_section = FALSE)`. When printing by section pressing the `<enter>` keypress will show the next section. Any other keypress will exit the interaction without printing any remaining sections. Color support is checked via `crayon::has_color()`. Customising rdoc output is possible via ‘options’, see [rdoc_options](#) for more details.

Examples

```
d <- system.file('extdata/rdoc_test.Rd', package = "rdoc")
rdoc::rdoc_rd(d)
```

rdoc_style	<i>rdoc style options</i>
------------	---------------------------

Description

Set Rd output formatting styles to be supplied to `options(rdoc.style = rdoc_style())`

Usage

```
rdoc_style(title = crayon::bold, section_titles = crayon::underline,  
arguments = NULL, code_style = prettycode::default_style())
```

Arguments

title	topic title.
section_titles	section titles
arguments	argument parameters.
code_style	code sections (e.g. examples) and inline code styles to be passed on to <code>prettycode::highlight</code>

Details

all inputs should be functions or NULL with the exception of `code_style` which should be a list of functions.

Value

A list

Examples

```
options(rdoc.style = rdoc::rdoc_style(arguments = crayon::italic))
```

rdoc_text	<i>Colourised R documentation</i>
-----------	-----------------------------------

Description

Access the underlying rdoc text lines

Usage

```
rdoc_text(topic, package = NULL, lib.loc = NULL)
```

Arguments

topic	character(1), help topic
package	character(1), package of help topic. Defaults to NULL
lib.loc	character(1), library location of package. Defaults to NULL

Value

character

Examples

```
txt <- rdoc::rdoc_text('min')
```

rdoc_text_formats *Rdoc text formatting*

Description

Set Rd text format defaults to be supplied to options(`rdoc.text_formats = rdoc_text_formats()`)

Usage

```
rdoc_text_formats(r_logo = crayon::combine_styles("bold", "blue"),
  pkg = function(x) paste0("{", x, "}"),
  inline_code = prettycode::default_style(), italic = crayon::italic,
  bold = crayon::bold, table = list(float = "center", border_style =
  "single", align = "left"), email = crayon::underline,
  url = crayon::underline, href = crayon::underline,
  link = function(x) x)
```

Arguments

r_logo	R symbol
pkg	pkg references
inline_code	inline code, options to pass to the style argument of <code>prettycode::highlight</code> as a list of functions.
italic	<i>italic</i> font
bold	bold font
table	tabular, options to pass to <code>cli::boxx</code> as a list.
email	email formats
url	url references
href	href
link	links

Details

The input will generally be a function, unless otherwise indicated above. Functions must take and return a character vector. If set as NULL, the relevant text will be displayed using the defaults of `tools::Rd2txt` or the calling function it is passed to. Note that `href` and `link` formatting refer to the source reference which will otherwise be excluded by `tools::Rd2txt`.

`use_rdoc`*Base replacements*

Description

rdoc replacements for ? and help

Usage

```
use_rdoc()
```

```
rm_rdoc()
```

Details

Calling `use_rdoc()` will override `utils::`?`` with `rdoc_question` and `utils::help` with `rdoc`. These replacements can be unset by calling `rm_rdoc()`.

Examples

```
rdoc::use_rdoc()
?help
rdoc::rm_rdoc()
?help
```

Index

* **rdoc access**

rdoc, [2](#)

* **use_rdoc shim**

use_rdoc, [7](#)

name, [3](#)

rdoc, [2](#)

rdoc_arguments (rdoc), [2](#)

rdoc_details (rdoc), [2](#)

rdoc_examples (rdoc), [2](#)

rdoc_options, [2](#), [3](#), [4](#)

rdoc_options-package (rdoc_options), [3](#)

rdoc_question, [3](#)

rdoc_rd, [4](#)

rdoc_style, [3](#), [5](#)

rdoc_text, [5](#)

rdoc_text_formats, [3](#), [6](#)

rdoc_usage (rdoc), [2](#)

rm_rdoc (use_rdoc), [7](#)

use_rdoc, [7](#)