

Package: rdocsyntax (via r-universe)

September 19, 2026

Title Syntax Highlighting for R HTML Documentation

Version 0.6.2.9000

Description Provides syntax highlighting for R HTML documentation.

License GPL (>= 3)

URL <https://github.com/kiendang/rdocsyntax>

BugReports <https://github.com/kiendang/rdocsyntax/issues>

Depends R (>= 2.10)

Imports xml2

Suggests V8, memoise, testthat (>= 3.0.0), withr

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Config/testthat/edition 3

Config/pak/sysreqs libxml2-dev

Repository <https://kiendang.r-universe.dev>

Date/Publication 2024-05-12 11:16:50 UTC

RemoteUrl <https://github.com/kiendang/rdocsyntax>

RemoteRef HEAD

RemoteSha 78ccd8b73163b76fe237191464274fd7e482fadd

Contents

highlight_html_docs	2
Index	4

highlight_html_docs	<i>Toggle syntax highlighting in R HTML help pages</i>
---------------------	--

Description

Toggle syntax highlighting in R HTML help pages

Usage

```
highlight_html_docs()
```

```
unhighlight_html_docs()
```

Details

R HTML help pages are rendered and served by the [internal help server](#) httpd. `highlight_html_docs()` enables syntax highlighting for them by replacing the built-in httpd with one that receives the original HTML doc, add syntax highlighting to the code examples inside, then serves the modified HTML.

The syntax highlighter in use comes from the [Ace text editor](#), the same editor which [Rstudio](#) uses.

To disable syntax highlighting and revert to the original help server, use `unhighlight_html_docs()`.

In case RStudio is not running, *e.g.* R console in a terminal is used instead, either set option `help_type` to "html" `options(help_type = "html")` to view all help pages in HTML mode by default. To view one single help page in HTML mode without changing the `help_type` option globally, use [help\(\)](#), *e.g.* `help(try, help_type = "html")`.

Value

Called for side effects. No return value.

Themes

When running inside RStudio, syntax highlighting always follows RStudio theme.

When running outside RStudio, theme can be changed by setting the `rdocsyntax.theme` option. Valid choices are:

- Light themes: `chrome`, `cloud9_day`, `clouds`, `crimson_editor`, `dawn`, `dreamweaver`, `eclipse`, `github`, `iplastic`, `katzenmilch`, `kuroir`, `solarized_light`, `sqlserver`, `textmate`, `tomorrow`, `xcode`
- Dark themes: `ambiance`, `chaos`, `cloud9_night`, `cloud9_night_low_color`, `clouds_midnight`, `cobalt`, `dracula`, `gob`, `gruvbox`, `idle_fingers`, `kr_theme`, `merbivore`, `merbivore_soft`, `mono_industrial`, `monokai`, `nord_dark`, `one_dark`, `pastel_on_dark`, `solarized_dark`, `terminal`, `tomorrow_night`, `tomorrow_night_blue`, `tomorrow_night_bright`, `tomorrow_night_eighties`, `twilight`, `vibrant_ink`

The default theme is `textmate` in case `rdocsyntax.theme` is not set or set to an invalid value.

Syntax highlighting for vignettes

Due to security reasons, by default only code in object documentation, as accessed by `?` or `help()`, is syntax highlighted. Syntax highlighting for other pages served by the dynamic help server `httpd` including vignettes can be enabled by setting `options(rdocsyntax.extra = TRUE)`.

Code in most vignettes has already been highlighted by default without the need for `rdocsyntax`. However one might still want to enable `rdocsyntax` for them anyway so that the color scheme matches that of RStudio. This is particularly useful when RStudio is set to a dark theme since most vignettes highlight their code with the `textmate` color scheme, which is a light theme and not dark mode friendly.

Setting `rdocsyntax.extra = TRUE` does not affect user defined `httpd` endpoints under `/custom/`, only those under `/doc/` and `/library/`.

Client side and server side highlighting

The code that does the highlighting is written in JavaScript since it depends on the <https://github.com/ajaxorg/ace/blob/v1.4.13>. Previously, up to `v0.5.x`, highlighting was done *server side*, i.e. `rdocsyntax` used `V8` to execute the JavaScript code and finished highlighting the HTML doc before returning it through `httpd`.

Starting from `v0.6.0`, highlighting has instead been done *client side*. The JavaScript highlighting code is injected into the original HTML doc in a `script` tag and then executed by whatever browser that eventually displays the doc, either RStudio or an external browser. Compared to server side highlighting, this is more efficient since `rdocsyntax` no longer has to run its own JavaScript engine. As a result, the heavy dependency on `V8` has been dropped.

Legacy server side highlighting is still available on an opt-in basis by setting `options(rdocsyntax.server_side_highlighting = TRUE)`. `V8` needs to be installed for this to work.

Examples

```
## Not run:
# Enable syntax highlighting
highlight_html_docs()

# Code in HTML documents is now highlighted
?try
# or if help pages are not displayed in HTML mode by default,
# e.g. when R is not running inside RStudio
help(try, help_type = "html")

# Switch to dracula theme (only takes effect outside of RStudio)
options(rdocsyntax.theme = "dracula")

# Disable syntax highlighting
unhighlight_html_docs()
## End(Not run)
```

Index

?, [3](#)

help(), [2](#), [3](#)

highlight_html_docs, [2](#)

internal help server, [2](#)

unhighlight_html_docs
 (highlight_html_docs), [2](#)