

Package ‘glossary’

May 8, 2026

Title Glossaries for Markdown and Quarto Documents

Date 2023-05-30

Version 1.0.0

Description Add glossaries to markdown and quarto documents by tagging individual words. Definitions can be provided inline or in a separate file.

License CC BY 4.0

Encoding UTF-8

RoxygenNote 7.2.3

Imports kableExtra, knitr, markdown, rvest, xml2, yaml

Suggests covr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://github.com/debruine/glossary>,
<https://debruine.github.io/glossary/>

BugReports <https://github.com/debruine/glossary/issues>

VignetteBuilder knitr

NeedsCompilation no

Author Lisa DeBruine [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0002-7523-5539>>)

Maintainer Lisa DeBruine <debruine@gmail.com>

Repository CRAN

Date/Publication 2023-05-30 18:00:02 UTC

Contents

glossary	2
glossary_add	3
glossary_path	4
glossary_popup	4
glossary_reset	5
glossary_style	5
glossary_table	6

glossary	<i>Display glossary entry</i>
----------	-------------------------------

Description

Display a glossary term with an optional popup of the definition, and add the term to the table created by [glossary_table](#). This function is mainly meant to be used via inline R in R Markdown or quarto documents, e.g.:

```
`r glossary("Alpha")` does not always have to equal .05.
```

Usage

```
glossary(
  term,
  display = term,
  def = NULL,
  add_to_table = TRUE,
  show = c("term", "def"),
  popup = glossary_popup(),
  path = glossary_path()
)
```

Arguments

term	The glossary term to link to, can contain spaces
display	The text to display (if different than the term)
def	The short definition to display on hover and in the glossary table; if NULL, this will be looked up from the file in the path argument
add_to_table	whether to add to the table created by glossary_table
show	whether to show the term or just the definition
popup	whether to show the popup on "click" or "hover" (or "none"); set default with glossary_popup
path	the path to the glossary file, or NULL for local definitions; set default with glossary_path

Details

If the path is set to "psyteachr", the glossary term will link to the [PsyTeachR glossary](#). Set show = "def" to just show the definition.

Value

character string

Examples

```
# set glossary path to example file
path <- system.file("glossary.yml", package = "glossary")
glossary_path(path)

glossary("alpha")
glossary("alpha", "$\\alpha$")
glossary("alpha", def = "The first letter of the Greek alphabet")
glossary("alpha", show = "term")
glossary("alpha", show = "def")
```

glossary_add	<i>Add a definition</i>
--------------	-------------------------

Description

Write a term and definition to an existing glossary file.

Usage

```
glossary_add(term, def, path = glossary_path(), replace = FALSE)
```

Arguments

term	The term to define
def	The definition to add
path	the path to the glossary file; set default with glossary_path
replace	Whether to replace an existing definition

Value

NULL; Called for side effects

Examples

```
# make a new glossary file
path <- tempfile("glossary", fileext = ".yml")
glossary_path(path, create = TRUE)

# add an entry for "joins"
glossary_add("joins", "Ways to combine data from two tables")

# now you can access the definition
glossary("joins")
```

glossary_path *Set or get the default glossary path*

Description

Set or get the default glossary path

Usage

```
glossary_path(path, create = FALSE)
```

Arguments

path	the path to the glossary file, or NULL for local definitions
create	create a new glossary file if it doesn't exist

Value

path string if path is NULL

Examples

```
path <- glossary_path() # get current path

# create (if doesn't exist) and set path
newpath <- tempfile("glossary", fileext = ".yml")
glossary_path(newpath, create = TRUE)

# set path (assumes file exists)
glossary_path(path)
```

glossary_popup *Set or get the default popup type*

Description

Set or get the default popup type

Usage

```
glossary_popup(popup)
```

Arguments

popup	If NULL, get the current default popup type, or set to one of "click", "hover", or "none"
-------	---

Value

string if popup is NULL

Examples

```
# get current popup style
popstyle <- glossary_popup()

# change popup to click style
glossary_popup("click")

# change back to original popup style
glossary_popup(popstyle)
```

glossary_reset	<i>Reset glossary table</i>
----------------	-----------------------------

Description

Resets the list that collects glossary entries for the table.

Usage

```
glossary_reset()
```

Value

NULL; Called for side effects

Examples

```
glossary_reset()
```

glossary_style	<i>Create CSS styles for glossary entries</i>
----------------	---

Description

Set the color and style of the linked in-text terms and pop-up definitions. Colors should be a valid CSS color string, such as "purple" or "#FF0000".

Usage

```
glossary_style(
  color = "purple",
  text_decoration = "underline",
  def_bg = "#333",
  def_color = "white"
)
```

Arguments

color	Text color of the linked term
text_decoration	Style of the linked term; a valid CSS text-decoration string, such as "none", "underline" or "red wavy underline"
def_bg	Background color of the definition pop-up
def_color	Text color of the definition pop-up

Value

A CSS style string

Examples

```
glossary_style("#003366", "underline")
```

glossary_table	<i>Display glossary table</i>
----------------	-------------------------------

Description

All terms defined with `glossary` (since the last call to `glossary_reset`) are added to a list, which this function displays using `kable` (or outputs as a data frame).

Usage

```
glossary_table(as_kable = TRUE)
```

Arguments

as_kable	if the output should be a <code>kableExtra</code> table or a data frame
----------	---

Value

kable table or data frame

Examples

```
glossary_reset()  
# add a definition to the table  
glossary("term", def = "definition", path = NULL)
```

```
glossary_table() # show table as kable  
glossary_table(FALSE) # or as a data frame
```

Index

[glossary](#), [2](#), [6](#)
[glossary_add](#), [3](#)
[glossary_path](#), [2](#), [3](#), [4](#)
[glossary_popup](#), [2](#), [4](#)
[glossary_reset](#), [5](#), [6](#)
[glossary_style](#), [5](#)
[glossary_table](#), [2](#), [6](#)