Package 'testthatdocs'

October 21, 2025

Type Package

Title Automated and Idempotent Unit Tests Documentation for
Reproducible Quality Assurance

Version 1.0.23

Description Automates documentation of test_that() calls within R test files.
The package scans test sources, extracts human-readable test titles (even when
composed with functions like paste() or glue::glue(), ... etc.), and generates
reproducible roxygen2-style listings that can be inserted both globally and
per-section. It ensures idempotent updates and supports customizable numbering
templates with hierarchical indices. Designed for developers, QA teams, and package
maintainers seeking consistent, self-documenting test inventories.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.5)

Suggests testthat (>= 3.0.0), glue, knitr, rmarkdown, withr

VignetteBuilder knitr

URL https://github.com/urniaz/testthatdocs

BugReports https://github.com/urniaz/testthatdocs/issues

RoxygenNote 7.3.2

Config/testthat/edition 3

NeedsCompilation no

Author Rafal Urniaz [aut, cre] (ORCID:
<https://orcid.org/0000-0003-0192-2165>)

Maintainer Rafal Urniaz <rafal.urniaz@cantab.net>

Repository CRAN

Date/Publication 2025-10-21 17:50:02 UTC

Contents

document   *Recursively generate test listings across a testthat tree*

Description

Walks the tests/testthat directory (by default), finds test files, and runs [document_file](#) on each
matching file. All options from document_file() are available here as pass-through parameters.

Usage

```
document(
  root = "tests/testthat",
  pattern = "^[Tt]est.*\\.[rR]$",
  recurse = TRUE,
  exclude = c("testthat.R"),
  section_prefix = "# -",
  template = c("simple", "advanced", "custom"),
  global_fmt = NULL,
  section_fmt = NULL,
  encoding = "UTF-8",
  backup = TRUE,
  write = TRUE,
  quiet = FALSE
)
```

Arguments

root	Character. Root directory to search. Default "tests/testthat".
pattern	Regular expression used with list.files(..., pattern=) to select test files. Default matches typical testthat files: "^[Tt]est.*\.[rR]$".
recurse	Logical. Whether to search subdirectories recursively. Default TRUE.
exclude	Character vector of basenames to exclude (e.g., "testthat.R"). Default c("testthat.R").
section_prefix	Character scalar. Lines starting with this prefix denote sections and are converted to #' @testsSection. Default "# -".
template	One of c("simple", "advanced", "custom"). Controls the default numbering format(s). Default "simple".
global_fmt	Character. Numbering template for the global listing. Uses placeholders {g}, {s}, {i}, {l}. If NULL, it is derived from template.
section_fmt	Character. Numbering template for section listings. If NULL, it is derived from template.
encoding	File encoding for reading and writing. Default "UTF-8".
backup	Logical. If TRUE, save a timestamped backup before overwriting. Default TRUE.
write	Logical. If TRUE, write changes back to path. If FALSE, return the would-be modified text without writing. Default TRUE.
quiet	Logical. If FALSE (default), prints progress messages.

Value

A list with components:

- `files`: character vector of processed file paths
- `results`: named list of `tests_listing_result` objects per file
- `listing`: combined data frame with a `file` column
- `backups`: character vector of backup paths (for files that were written)

Examples

```
## Not run:
  all_res <- document(
    root = "tests/testthat",
    template = "advanced",
    backup = TRUE,
    write = TRUE
  )
  head(all_res$listing)

## End(Not run)
```

---

document_file   *Generate idempotent listings of test_that() titles with sections*

Description

Scans an R text file for `test_that()` calls, generates a global listing and per-section listings as roxygen comments, and inserts them right after the corresponding markers. The function is **idempotent**: it only replaces content between existing @testsList and @testsSection markers and leaves other code/comments unchanged. If section headers are given using a plain-text prefix (e.g., "# -"), they are converted to roxygen markers #' @testsSection (with any following text treated as the section title).

Usage

```
document_file(
  path,
  section_prefix = "# -",
  template = c("simple", "advanced", "full", "custom"),
  global_fmt = NULL,
  section_fmt = NULL,
  encoding = "UTF-8",
  backup = TRUE,
  write = TRUE
)
```

Arguments

`path`	Character. Path to the text file to process (typically a test file).
`section_prefix`	Character scalar. Lines starting with this prefix denote sections and are converted to #' @testsSection. Default "# -".
`template`	One of c("simple", "advanced", "custom"). Controls the default numbering format(s). Default "simple".
`global_fmt`	Character. Numbering template for the global listing. Uses placeholders {g}, {s}, {i}, {l}. If NULL, it is derived from template.
`section_fmt`	Character. Numbering template for section listings. If NULL, it is derived from template.
`encoding`	File encoding for reading and writing. Default "UTF-8".
`backup`	Logical. If TRUE, save a timestamped backup before overwriting. Default TRUE.
`write`	Logical. If TRUE, write changes back to path. If FALSE, return the would-be modified text without writing. Default TRUE.

Details

The title extracted from `test_that()` is the first argument as a *raw expression*. If that argument is a single, quoted string (single/double/backtick), the outer quotes are stripped for cleaner listings. If it is constructed via functions like `paste()` or `glue::glue()`, the raw expression is listed without evaluation (and inner quotes remain).

Numbering is customizable via templates using placeholders:

- `{g}` – global incremental index across all tests
- `{s}` – section index (1-based, order of appearance)
- `{i}` – local index within a section (1-based)
- `{l}` – line index (line number in the final, modified text)
- aliases: `{local}` → `{i}`, `{line}` → `{l}`

Two presets are available via `template`:

- "simple" → "{g}"
- "advanced" → "{g}.{s}.{i}"
- "full" → "{g}.{s}.{i}.{l}"

You may fully override formats using `global_fmt` and `section_fmt`.

After inserting listings, the file is rescanned to compute the **final** `test_that()` line numbers reported in the returned data frame.

Value

A list with components:

- `text`: the final modified text (character vector, one element per line)
- `listing`: data frame of discovered tests with columns g, s, i, l (final line), title_raw, section_title.
- `written`: logical, whether the file was written
- `backup`: path to backup file (or NULL)

Examples

```
## Not run:
  res <- document_file("tests/testthat/test-example.R",
                              section_prefix = "# -",
                              template = "advanced")
  res$listing

## End(Not run)
```

Index