Package ‘printtree’

January 30, 2026

Type Package

Title Print Directory Trees for R Projects and Folders

Version 0.2.0

Description Provides tools to print a compact, readable directory tree for a folder or project. The package can automatically detect common project roots (e.g., 'RStudio' '.Rproj' files) and formats output for quick inspection of code and data organization. It supports typical tree customizations such as limiting depth, excluding files using ignore patterns, and producing clean, aligned text output suitable for console use, reports, and reproducible documentation. A snapshot helper can also render the tree output to a PNG image for sharing in issues, teaching material, or project documentation.

License MIT + file LICENSE

URL `https://github.com/PrigasG/printtree`

BugReports `https://github.com/PrigasG/printtree/issues`

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

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

NeedsCompilation no

Author George Arthur [aut, cre, cph]

Maintainer George Arthur <`prigasgenthian48@gmail.com`>

Repository CRAN

Date/Publication 2026-01-30 10:50:07 UTC

Contents

print_rtree *Print an R Project or Directory Tree*

Description

Prints a directory tree for a given path. Optionally detects an RStudio project (`.Rproj`) and can print from a project root.

Usage

```
print_rtree(
  path = NULL,
  ignore = c("renv", ".git", ".Rproj.user", "__pycache__", ".DS_Store", "node_modules",
    ".Rhistory"),
  max_depth = NULL,
  show_hidden = FALSE,
  project = c("auto", "root", "none"),
  search_paths = c(".", "..", "~/Documents", "~/Projects"),
  root_markers = c(".Rproj", "DESCRIPTION"),
  format = c("ascii", "unicode"),
  return_lines = FALSE,
  snapshot = FALSE,
  snapshot_file = "tree.png",
  snapshot_width = 800,
  snapshot_bg = c("white", "black"),
  snapshot_path = "."
)
```

Arguments

`path` Character. Directory path, project name, or `.Rproj` file. If NULL, uses ".".

`ignore` Character vector. Basenames to exclude (e.g., ".git", "renv").

`max_depth` Integer. Maximum depth to traverse. NULL for unlimited.

`show_hidden` Logical (TRUE/FALSE). Whether to include hidden files/directories (starting with ".").

`project` One of "auto", "root", "none".

- "auto": use `path` as-is (only resolves project name / .Rproj file)
- "root": walk upward from `path` to find a project root (via `root_markers`) and use it if found
- "none": never attempt root detection; print the tree from `path`

`search_paths` Character vector. Used only when `path` is not an existing directory (treated as a project name). Paths are searched in order.

`root_markers` Character vector. Markers used when `project = "root"` to detect a root directory. Special value ".Rproj" means "any file ending in .Rproj". Common markers include "DESCRIPTION" (R package root) and "_quarto.yml" (Quarto project root).

`format` One of "ascii" or "unicode". "ascii" is portable for all terminals.

`return_lines` Logical. If TRUE, invisibly return the printed character vector of lines.

`snapshot` Logical. If TRUE, saves a PNG snapshot of the tree.

`snapshot_file` Character. Snapshot PNG filename (or full path).

`snapshot_width` Integer. PNG width in pixels.

`snapshot_bg` One of "white" or "black" for snapshot background.

`snapshot_path` Character. Directory to save snapshot_file in when snapshot_file is not absolute.

Value

Invisible NULL, or a character vector of printed lines if `return_lines = TRUE`.

Examples

```
demo <- file.path(tempdir(), "printtree-demo")
if (dir.exists(demo)) unlink(demo, recursive = TRUE)
dir.create(demo, recursive = TRUE)
dir.create(file.path(demo, "R"))
file.create(file.path(demo, "R", "hello.R"))
file.create(file.path(demo, "README.md"))

print_rtree(demo)
print_rtree(demo, max_depth = 1)

png_file <- tempfile(fileext = ".png")
print_rtree(demo, snapshot = TRUE, snapshot_file = png_file)
```

Index