Package 'distill'

October 6, 2023

Type Package

Title 'R Markdown' Format for Scientific and Technical Writing

Version 1.6

Description Scientific and technical article format for the web.
'Distill' articles feature attractive, reader-friendly typography,
flexible layout options for visualizations, and full support for
footnotes and citations.

License Apache License 2.0

URL <https://github.com/rstudio/distill>,
<https://pkgs.rstudio.com/distill>

BugReports <https://github.com/rstudio/distill/issues>

Imports base64enc, bookdown (>= 0.8), digest, downlit (>= 0.4.1),
htmltools, jsonlite (>= 1.3), knitr (>= 1.15.6), lubridate (>=
1.7.10), mime, openssl, png, rmarkdown (>= 2.11), rprojroot,
rstudioapi, stats, stringr, tools, utils, whisker, xfun (>=
0.18), xml2, yaml

Suggests covr, DiagrammeR, htmlwidgets, progress, r2d3, rsconnect,
shiny, testthat (>= 3.0.0), withr

Config/Needs/website rstudio/quillt, pkgdown, tidyverse/tidytemplate

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.2.3

NeedsCompilation no

Author Christophe Dervieux [aut, cre]
(<https://orcid.org/0000-0003-4474-2498>),
JJ Allaire [aut] (<https://orcid.org/0000-0003-0174-9868>),
Rich Iannone [aut],
Alison Presmanes Hill [aut] (<https://orcid.org/0000-0002-8082-1890>),
Yihui Xie [aut] (<https://orcid.org/0000-0003-0645-5666>),

Posit Software, PBC [cph, fnd],
Google LLC [ctb, cph] (https://distill.pub/guide/),
Nick Williams [ctb, cph] (https://wicky.nillia.ms/headroom.js/),
Denis Demchenko [ctb, cph] (https://github.com/lancedikson/bowser),
The Polymer Authors [ctb, cph]
 (https://www.webcomponents.org/polyfills/),
Gábor Csárdi [ctb, cph] (whoami),
JooYoung Seo [ctb] (<https://orcid.org/0000-0002-4064-6012>)

R topics documented:

---

create_article	*Create a new article*

Description

Create (and optionally edit) a new distill article.

Usage

```
create_article(
  file,
  template = "distill_article",
  package = "distill",
  create_dir = FALSE,
  edit = TRUE
)
```

Arguments

`file`	File name for the draft
`template`	Template to use as the basis for the draft. This is either the full path to a template directory or the name of a template directory within the `rmarkdown/templates` directory of a package.
`package`	(Optional) Name of package where the template is located.
`create_dir`	TRUE to create a new directory for the document (defaults to FALSE).
`edit`	TRUE to edit the template immediately

`create_post`	*Create a new blog post*

Description

Create a new blog post

Usage

```
create_post(
  title,
  collection = "posts",
  author = "auto",
  slug = "auto",
  date = Sys.Date(),
  date_prefix = date,
  draft = FALSE,
  edit = interactive()
)
```

Arguments

`title`	Post title
`collection`	Collection to create the post within (defaults to "posts")
`author`	Post author. Automatically drawn from previous post if not provided.
`slug`	Post slug (directory name). Automatically computed from title if not provided.
`date`	Post date (defaults to current date)
`date_prefix`	Date prefix for post slug (preserves chronological order for posts within the filesystem). Pass NULL for no date prefix.
`draft`	Mark the post as a `draft` (don't include it in the article listing).
`edit`	Open the post in an editor after creating it.

Note

This function must be called from with a working directory that is within a Distill website.

Examples

```
## Not run:
library(distill)
create_post("My Post")

## End(Not run)
```

---

create_theme   *Create a Distill theme CSS file*

Description

Create a theme CSS file and write it to the current working directory

Usage

```
create_theme(name = "theme", edit = TRUE)
```

Arguments

name   Name of theme file (will be written as name.css)
edit   Open an editor for the theme file

Details

How do you apply a custom theme to a Distill site or blog? You have two options:

1. Apply it site-wide by adding a theme key to the top-level of your _site.yml configuration file:

```
name: "distill"
title: "Distill for R Markdown"
theme: theme.css
navbar:
  # (navbar definition here)
```

2. Apply to an individual article by adding a theme key to your article's YAML front-matter:

```
---
title: "The Sharpe Ratio"
output:
  distill::distill_article:
    toc: true
    theme: theme.css
---
```

The second option allows for you to apply your theme to individual articles, while using a different theme for the rest of your site. Note that this is only possible for stand-alone articles within a website — you cannot apply a theme to individual blog posts *only*.

More details

For further details about theming refer to the online documentation.

---

create_website *Create a Distill website*

Description

Create a basic skeleton for a Distill website or blog. Use the `create_website()` function for a website and the `create_blog()` function for a blog.

Usage

```
create_website(dir, title, gh_pages = FALSE, edit = interactive())

create_blog(dir, title, gh_pages = FALSE, edit = interactive())
```

Arguments

dir	Directory for website
title	Title of website
gh_pages	Configure the site for publishing using GitHub Pages
edit	Open site index file or welcome post in an editor.

Note

The `dir` and `title` parameters are required (they will be prompted for interactively if they are not specified).

Examples

```
## Not run:
library(distill)
create_website("mysite", "My Site")

## End(Not run)
```

---

distill_article   *R Markdown format for Distill articles*

Description

Scientific and technical writing, native to the web.

Usage

```
distill_article(
  toc = FALSE,
  toc_depth = 3,
  toc_float = TRUE,
  fig_width = 6.5,
  fig_height = 4,
  fig_retina = 2,
  fig_caption = TRUE,
  dev = "png",
  smart = TRUE,
  code_folding = FALSE,
  self_contained = TRUE,
  highlight = "default",
  highlight_downlit = TRUE,
  mathjax = "default",
  extra_dependencies = NULL,
  theme = NULL,
  css = NULL,
  includes = NULL,
  keep_md = FALSE,
  lib_dir = NULL,
  md_extensions = NULL,
  pandoc_args = NULL,
  ...
)
```

Arguments

`toc`	TRUE to include a table of contents in the output
`toc_depth`	Depth of headers to include in table of contents
`toc_float`	Float the table of contents to the left when the article is displayed at widths > 1000px. If set to FALSE or the width is less than 1000px the table of contents will be placed above the article body.
`fig_width`	Default width (in inches) for figures
`fig_height`	Default height (in inches) for figures

fig_retina	Scaling to perform for retina displays (defaults to 2, which currently works for all widely used retina displays). Set to NULL to prevent retina scaling. Note that this will always be NULL when keep_md is specified (this is because fig_retina relies on outputting HTML directly into the markdown document).
fig_caption	TRUE to render figures with captions
dev	Graphics device to use for figure output (defaults to png)
smart	Produce typographically correct output, converting straight quotes to curly quotes, --- to em-dashes, -- to en-dashes, and ... to ellipses.
code_folding	Include code blocks hidden, and allow users to optionally display the code by clicking a "Show code" button just above the output. Pass a character vector to customize the text of the "Show code" button. You can also specify code_folding as chunk option for per-chunk folding behavior.
self_contained	Produce a standalone HTML file with no external dependencies, using data: URIs to incorporate the contents of linked scripts, stylesheets, images, and videos. Note that even for self contained documents MathJax is still loaded externally (this is necessary because of its size).
highlight	Syntax highlighting style. Supported styles include "default", "rstudio", "tango", "pygments", "kate", "monochrome", "espresso", "zenburn", "breezedark", and "haddock". Pass NULL to prevent syntax highlighting.
highlight_downlit	Use the **downlit** package to highlight R code (including providing hyperlinks to function documentation).
mathjax	Include mathjax. The "default" option uses an https URL from a MathJax CDN. The "local" option uses a local version of MathJax (which is copied into the output directory). You can pass an alternate URL or pass NULL to exclude MathJax entirely.
extra_dependencies	Extra dependencies as a list of the html_dependency class objects typically generated by [htmltools:htmlDependency()]().
theme	CSS file with theme variable definitions
css	CSS and/or Sass files to include. Files with an extension of .sass or .scss are compiled to CSS via sass::sass().
includes	Named list of additional content to include within the document (typically created using the [includes]() function).
keep_md	Keep the markdown file generated by knitting.
lib_dir	Directory to copy dependent HTML libraries (e.g. jquery, bootstrap, etc.) into. By default this will be the name of the document with _files appended to it.
md_extensions	Markdown extensions to be added or removed from the default definition of R Markdown. See the [rmarkdown_format]() for additional details.
pandoc_args	Additional command line options to pass to pandoc
...	Additional function arguments to pass to the base R Markdown HTML output formatter [html_document_base]()

Details

Distill articles feature attractive, reader-friendly typography, flexible layout options for visualizations, and full support for footnotes and citations.

---

distill_website   *R Markdown site generator for Distill websites*

Description

R Markdown site generator for Distill websites

Usage

```
distill_website(input, encoding = getOption("encoding"), ...)
```

Arguments

input	Website directory (or the name of a file within the directory).
encoding	Ignored. The encoding is always assumed to be UTF-8.
...	Currently unused.

---

import_post   *Import a post into a blog*

Description

Import a distill post from an external source (e.g. GitHub repo, RPubs article, etc.). Importable posts must have `distill::distill_article` as the output format in the YAML.

Usage

```
import_post(
  url,
  slug = "auto",
  date = Sys.Date(),
  date_prefix = date,
  check_license = TRUE,
  overwrite = FALSE,
  view = interactive()
)

update_post(slug, view = interactive())
```

Arguments

`url`	URL for distill post to import
`slug`	Post slug (directory name). Automatically computed from title if not provided.
`date`	Post date (defaults to current date)
`date_prefix`	Date prefix for post slug (preserves chronological order for posts within the filesystem). Pass `NULL` for no date prefix.
`check_license`	Verify that the imported posted has a creative commons license
`overwrite`	Overwrite existing post? (defaults to `FALSE`, use [update_post()](#) to update an existing post in-place).
`view`	View the post after importing it.

Value

Returns (invisibly) a logical indicating whether the operation completed (it may not complete if, for example, the user chose not to import an article that lacked a creative commons license).

---

publish_website	*Publish a Distill Website*

Description

Publish a website to RStudio Connect

Usage

```
publish_website(
  site_dir = ".",
  site_name = NULL,
  method = c("rsconnect"),
  server = NULL,
  account = NULL,
  render = TRUE,
  launch_browser = interactive()
)
```

Arguments

`site_dir`	Directory containing website. Defaults to current working directory.
`site_name`	Name for the site (names must be unique within an account). Defaults to the name provided in `_site.yml` (or to the name of the site_dir if there is no `name` specified).
`method`	Publishing method (currently only "rsconnect" is available)

account, server

Uniquely identify a remote server with either your user account, the server name, or both. If neither are supplied, and there are multiple options, you'll be prompted to pick one.

Use accounts() to see the full list of available options.

render   TRUE to render the site locally before publishing.

launch_browser If TRUE, the system's default web browser will be launched automatically after the site is deployed. Defaults to TRUE in interactive sessions only.

Examples

```
## Not run:
library(distill)
publish_website()

## End(Not run)
```

rename_post_dir   *Rename a blog post directory*

Description

Rename a blog post directory, by default using the title and date specified within the post's front matter.

Usage

```
rename_post_dir(post_dir, slug = "auto", date_prefix = "auto")
```

Arguments

post_dir   Path to post directory

slug   Post slug (directory name). Automatically computed from title if not provided.

date_prefix   Date prefix for post. Defaults to the post's date field (or the current date if there is none). Pass NULL to not use a date prefix.

Note

This function must be called from with a working directory that is within a Distill website.

Examples

```
## Not run:
library(distill)
rename_post_dir("_posts/2020-09-12-my-post")
rename_post_dir("_posts/2020-09-12-my-post", date_prefix = "9/15/2020")

## End(Not run)
```

Index