

Package ‘liteformats’

August 5, 2026

Title Lightweight Output Formats for 'litedown'

Version 0.1.0

Description A collection of lightweight, minimalist output formats and templates for 'litedown' by Xie (2026) <[doi:10.32614/CRAN.package.litedown](https://doi.org/10.32614/CRAN.package.litedown)>, including resumes, cover letters, and other common document types. Documents are rendered with HTML and CSS and can be printed to PDF with a 'Chromium'-based browser, without requiring 'Pandoc' or a 'LaTeX' installation.

License MIT + file LICENSE

URL <https://nanx.me/liteformats/>,
<https://github.com/nanxstats/liteformats>

BugReports <https://github.com/nanxstats/liteformats/issues>

Encoding UTF-8

Imports litedown (>= 0.10), xfun (>= 0.60)

Suggests testit

VignetteBuilder litedown

SystemRequirements Chromium, Google Chrome, or Microsoft Edge (for PDF output)

RoxygenNote 8.0.0

NeedsCompilation no

Author Nan Xiao [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0002-0250-5673>>),
Yihui Xie [ctb, cph] (pages.js and pages.css libraries)

Maintainer Nan Xiao <me@nanx.me>

Repository CRAN

Date/Publication 2026-08-05 07:10:08 UTC

Contents

cover_letter	2
resume	4
use_resume	6

Index[7](#)

cover_letter	<i>Render a cover letter</i>
--------------	------------------------------

Description

Render a two-column cover letter to self-contained HTML or PDF. The narrow left column holds the sender's details, while the letter remains the visual foreground in the wider right column.

Usage

```
cover_letter(
  input,
  output = NULL,
  author = NULL,
  address = NULL,
  phone = NULL,
  email = NULL,
  website = NULL,
  date = NULL,
  greeting = NULL,
  font_family = NULL,
  google_font = NULL,
  font_files = NULL,
  font_size = NULL,
  font_scale = NULL,
  ligatures = NULL,
  line_height = NULL,
  paragraph_spacing = NULL,
  section_spacing = NULL,
  margins = NULL,
  paper = NULL,
  link_color = NULL,
  paged = NULL,
  keep_html = FALSE,
  browser = NULL,
  envir = parent.frame()
)
```

Arguments

input	Path to a .Rmd or .md source document.
output	Output path ending in .html or .pdf. If NULL, an HTML file is created beside input.
author, address, phone, email, website	Contact metadata. NULL uses the corresponding value in the source document's YAML header.

date, greeting	Letter date and salutation. NULL uses YAML metadata.
font_family	A CSS font-family value. For example, "Charter, Georgia, serif".
google_font	A Google Fonts family name or a full Google Fonts CSS URL. This is an explicit online resource and is never used by default. A family name loads its default face; pass a full URL plus the matching font_family when specific axes or weights are required.
font_files	A named character vector or list of local font files. Names may be regular, italic, bold, and bold_italic. Relative paths are resolved from the source document. Fonts are embedded in the HTML.
font_size	A CSS length such as "11pt".
font_scale	A positive number used to scale font_size. This is useful when replacing a font with different metrics.
ligatures	Whether to enable common and discretionary ligatures. Defaults to FALSE for more predictable text extraction from PDFs.
line_height	A positive unitless CSS line height or CSS length.
paragraph_spacing, section_spacing	CSS lengths controlling vertical rhythm.
margins	Page margins. Supply one to four CSS lengths using standard CSS shorthand order, or numeric values interpreted as inches.
paper	A paper name ("letter", "a4", or "legal") or a two-element vector containing width and height as CSS lengths.
link_color	Any valid CSS color.
paged	Whether the HTML preview should open in a paginated layout. Printing always invokes the packaged pages.js pagination.
keep_html	For PDF output, whether to retain the intermediate HTML beside the PDF.
browser	Path to Chromium, Google Chrome, or Microsoft Edge. NULL lets <code>xfun::browser_print()</code> discover it.
envir	Environment in which R code in input is evaluated.

Value

The normalized output path, invisibly.

Examples

```
source <- use_cover_letter(tempfile(fileext = ".Rmd"))
cover_letter(source, tempfile(fileext = ".html"))

## Not run:
# PDF output requires a local Chromium-based browser
cover_letter(source, "cover-letter.pdf")

## End(Not run)
```

 resume

Render a resume

Description

Render a Markdown or R Markdown resume to a self-contained HTML document or to PDF through a headless Chromium-based browser. The output format is inferred from the extension of output.

Usage

```
resume(
  input,
  output = NULL,
  author = NULL,
  job_title = NULL,
  address = NULL,
  phone = NULL,
  email = NULL,
  website = NULL,
  linkedin = NULL,
  font_family = NULL,
  google_font = NULL,
  font_files = NULL,
  font_size = NULL,
  font_scale = NULL,
  ligatures = NULL,
  line_height = NULL,
  paragraph_spacing = NULL,
  section_spacing = NULL,
  margins = NULL,
  paper = NULL,
  link_color = NULL,
  paged = NULL,
  keep_html = FALSE,
  browser = NULL,
  envir = parent.frame()
)
```

Arguments

input	Path to a .Rmd or .md source document.
output	Output path ending in .html or .pdf. If NULL, an HTML file is created beside input.
author, address, phone, email, website	Contact metadata. NULL uses the corresponding value in the source document's YAML header.

<code>job_title, linkedin</code>	Resume-specific professional and contact metadata.
<code>font_family</code>	A CSS font-family value. For example, "Charter, Georgia, serif".
<code>google_font</code>	A Google Fonts family name or a full Google Fonts CSS URL. This is an explicit online resource and is never used by default. A family name loads its default face; pass a full URL plus the matching <code>font_family</code> when specific axes or weights are required.
<code>font_files</code>	A named character vector or list of local font files. Names may be <code>regular</code> , <code>italic</code> , <code>bold</code> , and <code>bold_italic</code> . Relative paths are resolved from the source document. Fonts are embedded in the HTML.
<code>font_size</code>	A CSS length such as "11pt".
<code>font_scale</code>	A positive number used to scale <code>font_size</code> . This is useful when replacing a font with different metrics.
<code>ligatures</code>	Whether to enable common and discretionary ligatures. Defaults to <code>FALSE</code> for more predictable text extraction from PDFs.
<code>line_height</code>	A positive unitless CSS line height or CSS length.
<code>paragraph_spacing, section_spacing</code>	CSS lengths controlling vertical rhythm.
<code>margins</code>	Page margins. Supply one to four CSS lengths using standard CSS shorthand order, or numeric values interpreted as inches.
<code>paper</code>	A paper name ("letter", "a4", or "legal") or a two-element vector containing width and height as CSS lengths.
<code>link_color</code>	Any valid CSS color.
<code>paged</code>	Whether the HTML preview should open in a paginated layout. Printing always invokes the packaged <code>pages.js</code> pagination.
<code>keep_html</code>	For PDF output, whether to retain the intermediate HTML beside the PDF.
<code>browser</code>	Path to Chromium, Google Chrome, or Microsoft Edge. <code>NULL</code> lets <code>xfun::browser_print()</code> discover it.
<code>envir</code>	Environment in which R code in <code>input</code> is evaluated.

Details

Document metadata and typographic settings may be supplied in the YAML header. Renderer arguments take precedence over YAML. See `use_resume()` for a complete starter document.

Value

The normalized output path, invisibly.

Examples

```
source <- use_resume(tempfile(fileext = ".Rmd"))
resume(source, tempfile(fileext = ".html"))

## Not run:
```

```
# PDF output requires a local Chromium-based browser
resume(source, "resume.pdf")

# Embed local font files
resume(
  source,
  font_files = c(
    regular = "fonts/MySerif-Regular.woff2",
    italic = "fonts/MySerif-Italic.woff2",
    bold = "fonts/MySerif-Bold.woff2"
  ),
  font_scale = 0.96,
  ligatures = FALSE
)

## End(Not run)
```

use_resume

Create a resume or cover letter from a starter

Description

Copy an editable R Markdown starter document to path. Static assets used by the rendered formats remain inside the installed package; the copied source contains only Markdown content and YAML settings.

Usage

```
use_resume(path = "resume.Rmd", overwrite = FALSE)
```

```
use_cover_letter(path = "cover-letter.Rmd", overwrite = FALSE)
```

Arguments

path	Destination path.
overwrite	Whether to replace an existing file.

Value

The normalized destination path, invisibly.

Examples

```
use_resume(tempfile(fileext = ".Rmd"))
use_cover_letter(tempfile(fileext = ".Rmd"))
```

Index

`cover_letter`, [2](#)

`resume`, [4](#)

`use_cover_letter (use_resume)`, [6](#)

`use_resume`, [6](#)

`xfun::browser_print()`, [3](#), [5](#)