

Package ‘resumer’

July 23, 2025

Title Build Resumes with R

Version 0.0.5

Description Using a CSV, LaTeX and R to easily build attractive resumes.

Depends R (>= 3.2.1)

License BSD_3_clause + file LICENSE

LazyData true

ByteCompile true

URL <https://github.com/jaredlander/resumer>

BugReports <https://github.com/jaredlander/resumer/issues>

Suggests testthat

Imports useful, dplyr, rmarkdown

RoxygenNote 7.1.1

NeedsCompilation no

Author Jared Lander [aut, cre]

Maintainer Jared Lander <packages@jaredlander.com>

Repository CRAN

Date/Publication 2021-02-12 15:00:02 UTC

Contents

resumer-package	2
createJobFile	2
generateListing	3
generateMultipleListings	4
generateSection	5
jobs	6
resumer	7
Index	8

resumer-package	<i>resumer</i>
-----------------	----------------

Description

Using a CSV, LaTeX and R to easily build attractive resumes.

createJobFile	<i>createJobFile</i>
---------------	----------------------

Description

Creates a CSV to hold information about jobs and research

Usage

```
createJobFile(filename = "Resume.csv", sep = ",")
```

Arguments

filename	Name of file in which to create the csv
sep	Separator to use, ; is suggested

Details

This creates a data.frame and writes an empty file to disk. This file should either be edited by hand or with a data.frame.

Value

An empty data.frame

Author(s)

Jared P. Lander

Examples

```
## Not run:  
createJobFile()  
  
## End(Not run)
```

generateListing	<i>generateListing</i>
-----------------	------------------------

Description

Generate LaTeX code for job info

Usage

```
generateListing(data, bullets, type = "Job", specialChars = "&")
```

Arguments

data	data.frame holding the info for one job
bullets	The BulletName's for the desired rows
type	The type of subsection to build; defaults to 'Job', the other currently supported value is 'Research'
specialChars	Vector of characters that need to be double-backslashed escaped

Details

Given a subsetting dataset of just one job this generates LaTeX code. Given jobname and company name, print out the section.

Value

LaTeX code for a subsection in the resume

Author(s)

Jared P. Lander

See Also

[generateMultipleListings](#) [generateSection](#)

Examples

```
library(dplyr)

jobs <- read.csv(system.file('examples/Jobs.csv', package='resumer'))
oneJob <- jobs %>% filter(Company=='Pied Piper', JobName=='Tech Startup')
generateListing(oneJob)
generateListing(oneJob, bullets=c(1, 3))

oneResearch <- jobs %>% filter(JobName=='Oddie Research', Company=='Hudson University')
generateListing(oneResearch, bullets=4, type='Research')
generateListing(oneResearch, bullets=4:5, type='Research')
```

```
generateMultipleListings
      generateMultipleListings
```

Description

Generate an entire resume section

Usage

```
generateMultipleListings(data, jobList, type = "Job", specialChars = "&")
```

Arguments

data	data.frame holding the info for one job
jobList	A list of jobs, each of which is a list where the first element is the Company, the second is the JobName and the third is a vector of BulletName's
type	The type of section to build; defaults to 'Job', the other currently supported value is 'Research'
specialChars	Vector of characters that need to be double-backslashed escaped

Details

Using a list of lists to describe jobs generate text for each job subsection

Value

A vector of text, one for each job

Author(s)

Jared P. Lander

See Also

[generateListing](#) [generateSection](#)

Examples

```
jobList <- list(
  list("Pied Piper", "Tech Startup", c(1, 3)),
  list("Goliath National Bank", "Bank Intern", 1:3),
  list("Surveyors Inc", "Survey Stats", 1:2)
)

generateMultipleListings(jobs, jobList)
```

generateSection	<i>generateSection</i>
-----------------	------------------------

Description

Generate an entire job/research section

Usage

```
generateSection(  
  data,  
  jobList,  
  sectionName = "Relevant Experience",  
  type = "Job",  
  specialChars = "&"  
)
```

Arguments

data	data.frame holding the info for one job
jobList	A list of jobs, each of which is a list where the first element is the Company, the second is the JobName and the third is a vector of BulletName's
sectionName	Name to be printed at the top of the section
type	The type of section to build; defaults to 'Job', the other currently supported value is 'Research'
specialChars	Vector of characters that need to be double-backslashed escaped

Details

Given a jobs data.frame and a job list generate all the code needed for a jobs section

Value

All the text needed for a job section

Author(s)

Jared P. Lander

See Also

[generateListing](#) [generateMultipleListings](#)

Examples

```
data(jobs)
jobList <- list(
  list("Pied Piper", "Tech Startup", c(1, 3)),
  list("Goliath National Bank", "Bank Intern", 1:3),
  list("Surveyors Inc", "Survey Stats", 1:2)
)

generateSection(jobs, jobList)
```

jobs	<i>Prices of 50,000 round cut diamonds.</i>
------	---

Description

A dataset containing the listings for a resume

Usage

jobs

Format

A data frame with 27 rows and 10 variables:

- JobName** The internal name given to ID the job
- Company** Name of company
- Location** Job Location
- Title** Position Title
- Start** Start date of job
- End** End date of job
- Bullet** Bullet points for jobs
- BulletName** Name or ID for bullets
- Type** Type of job, either a job or research
- Description** Short blurb about the job

Source

Manufactured data

resumer	<i>resumer</i>
---------	----------------

Description

Convert to a resume

Usage

```
resumer(
  fig_width = 4,
  fig_height = 2.5,
  fig_crop = TRUE,
  dev = "pdf",
  highlight = "default",
  keep_tex = FALSE,
  latex_engine = "pdflatex",
  includes = NULL,
  md_extensions = NULL,
  pandoc_args = NULL,
  template = "default"
)
```

Arguments

<code>fig_width</code>	Default width (in inches) for figures
<code>fig_height</code>	Default height (in inches) for figures
<code>fig_crop</code>	TRUE to automatically apply the <code>pdfcrop</code> utility (if available) to pdf figures
<code>dev</code>	Graphics device to use for figure output (defaults to pdf)
<code>highlight</code>	Syntax highlighting style. Supported styles include "default", "tango", "pygments", "kate", "monochrome", "espresso", "zenburn", and "haddock". Pass NULL to prevent syntax highlighting.
<code>keep_tex</code>	Keep the intermediate tex file used in the conversion to PDF
<code>latex_engine</code>	LaTeX engine for producing PDF output. Options are "pdflatex", "lualatex", and "xelatex".
<code>includes</code>	Named list of additional content to include within the document (typically created using the includes function).
<code>md_extensions</code>	Markdown extensions to be added or removed from the default definition or R Markdown. See the rmarkdown_format for additional details.
<code>pandoc_args</code>	Additional command line options to pass to pandoc
<code>template</code>	Pandoc template to use for rendering. Pass "default" to use the resumer package default template; pass NULL to use pandoc's built-in template; pass a path to use a custom template that you've created.

Index

* **datasets**

jobs, [6](#)

createJobFile, [2](#)

generateListing, [3](#), [4](#), [5](#)

generateMultipleListings, [3](#), [4](#), [5](#)

generateSection, [3](#), [4](#), [5](#)

includes, [7](#)

jobs, [6](#)

resumer, [7](#)

resumer-package, [2](#)

rmarkdown_format, [7](#)