Package 'texor'

May 8, 2026

Type Package

Title Converting 'LaTeX' 'R Journal' Articles into 'RJ-web-articles'

Version 1.6.0

Description Articles in the 'R Journal' were first authored in 'LaTeX', which performs admirably for 'PDF' files but is less than ideal for modern online interfaces. The 'texor' package does all the transitional chores and conversions necessary to move to the online versions.

License MIT + file LICENSE

URL https://github.com/Abhi-1U/texor

BugReports https://github.com/Abhi-1U/texor/issues

Encoding UTF-8

RoxygenNote 7.3.2

Imports rmarkdown, pdftools, tools, tinytex, yaml, stringr, xfun, logger, rjtools, rebib, cli, whisker, desc

Suggests bookdown, knitr, spelling, rstudioapi, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Language en-US

SystemRequirements pandoc (>= 3.1) - https://pandoc.org

NeedsCompilation no

Author Abhishek Ulayil [aut, cre, cph] (ORCID: <https://orcid.org/0009-0000-6935-8690>), Heather Turner [ctb] (ORCID: <https://orcid.org/0000-0002-1256-3375>), Christophe Dervieux [ctb] (ORCID: <https://orcid.org/0000-0003-4474-2498>), Mitchell O'Hara-Wild [ctb] (ORCID: <https://orcid.org/0000-0001-6729-7695>), Dianne Cook [ctb] (ORCID: <https://orcid.org/0000-0002-3813-7155>), Yinxiang Huang [ctb] (ORCID: <https://orcid.org/0009-0007-2031-7901>)

Maintainer Abhishek Ulayil <perricoq@outlook.com>

Repository CRAN

Date/Publication 2025-09-08 18:50:02 UTC

Contents

`article_has_tikz` *Check if article has tikz images or not*

Description

This simple utiliy function will check for tikzpicture environment

Usage

```
article_has_tikz(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Value

TRUE if tikz image is present else FALSE

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::article_has_tikz(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

---

check_markdown_file *check markdown file*

Description

Checks if the markdown file generated is empty or not due to some pandoc related error during conversion to markdown.

Usage

```
check_markdown_file(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Value

FALSE if markdown file is corrupted/empty else TRUE

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
rmarkdown::pandoc_version()
texor::include_style_file(your_article_path)
rebib::aggregate_bibliography(your_article_path)
texor::convert_to_markdown(your_article_path)
texor::check_markdown_file(your_article_path)
unlink(your_article_folder, recursive = TRUE)
```

`convert_to_markdown` *convert LaTeX wrapper to markdown*

Description

Uses pandoc along with several lua filters found at inst/extdata/filters in texor package

Usage

```
convert_to_markdown(
  article_dir,
  kable_tab = TRUE,
  autonumber_eq = FALSE,
  fig_in_r = TRUE
)
```

Arguments

`article_dir`	path to the directory which contains tex article
`kable_tab`	converts to kable table instead of markdown tables
`autonumber_eq`	whether to autonumber the equations, default is FALSE
`fig_in_r`	whether to include figures in R code chunks, default is TRUE

Details

convert latex(wrapper) file to markdown

Value

creates a converted markdown file, as well as a pkg_meta.yaml file

Note

pandoc (along with lua interpreter) is already installed with R-studio, hence if not using R-studio you will need to install pandoc. https://pandoc.org/installing.html

Use pandoc version greater than or equal to 3.1

Kable tables will work for simple static data, any math / code / image within any table will send the package into fallback mode (normal markdown tables) for the rest of tables in the article.

Examples

```
# Note This is a minimal example to execute this function
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
```

```
rmarkdown::pandoc_version()
texor::include_style_file(your_article_path)
rebib::aggregate_bibliography(your_article_path)
texor::convert_to_markdown(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

`convert_to_native` *convert LaTeX wrapper to native pandoc AST*

Description

Uses pandoc along with several lua filters found at inst/extdata/filters in texor package

Usage

```
convert_to_native(article_dir, autonumber_eq = FALSE)
```

Arguments

`article_dir`	path to the directory which contains tex article
`autonumber_eq`	whether to autonumber the equations, default is FALSE

Details

convert latex(wrapper) file to pandoc AST

Value

creates a converted native AST file, as well as a pkg_meta.yaml file

Note

pandoc (along with lua interpreter) is already installed with R-studio, hence if not using R-studio you will need to install pandoc. https://pandoc.org/installing.html

Use pandoc version greater than or equal to 3.1

Examples

```
# Note This is a minimal example to execute this function
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
rmarkdown::pandoc_version()
texor::include_style_file(your_article_path)
rebib::aggregate_bibliography(your_article_path)
texor::convert_to_native(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

convert_to_png *convert one single pdf file to png*

Description

function to invoke ‘pdftools:pdf_convert()‘

This function is designed to be used internally and is called by ‘texor::pdf_to_png(file_dir)‘ function for converting individual of pdf image.

Note : The extensions in LaTeX source code will automatically be changed during pandoc conversion by a lua filter (refer : inst/extdata/image_filter.lua)

Usage

```
convert_to_png(file_path, dpi = 180)
```

Arguments

file_path	path to the pdf file
dpi	Set DPI for converting PDF files. default: 180

Value

png file of the same

Note

If you find inconsistencies in the raster image generated from PDF using this function. Please update poppler utils to newer versions (possibly latest one).

Examples

```
article_dir <- system.file("examples/pdf_conversion",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"pdf_conversion",sep="/")
rmarkdown::pandoc_version()
texor::convert_to_png(paste0(your_article_path,"/normal.pdf"))
unlink(your_article_folder,recursive = TRUE)
```

copy_other_files *Copy Supporting Documents like images,bib file,etc.*

Description

Copies supporting documents like images,pdf,bib files into the output folder for building the HTML version of the R-Markdown file.

Usage

```
copy_other_files(from_path)
```

Arguments

from_path	String indicating base path for the working directory

Value

copies dependency files into the output folder.

Examples

```
article_dir <- system.file("examples/article", package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
rmarkdown::pandoc_version()
texor::include_style_file(your_article_path)
rebib::aggregate_bibliography(your_article_path)
texor::copy_other_files(your_article_path)
list.files(paste0(your_article_path,"/web/"))
unlink(your_article_folder,recursive = TRUE)
```

count_env *count latex environments*

Description

count common environments like table,figure,verbatim etc..

Usage

```
count_env(article_dir, env_name)
```

Arguments

article_dir	path to the directory which contains RJ article
env_name	name of the environment

Value

count of the environment, FALSE otherwise

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
figures <- texor::count_env(article_dir, "figure")
print(paste("figure count : ", figures))
```

`count_inline` *count inline elements*

Description

counts inline elements embedded within the latex file currently supported inlines : math (based on $$), code (based on \code) and Citations (based on \cite,\citealp, \citep, \citet)

Usage

```
count_inline(article_dir, inline)
```

Arguments

`article_dir`	path to the directory which contains RJ article
`inline`	name of the inline element

Value

count of the inline element, FALSE otherwise

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
math <- texor::count_inline(article_dir, "math")
code <- texor::count_inline(article_dir, "inlinecode")
cite <- texor::count_inline(article_dir, "cite")
print(paste("math inlines : ", math, "\n",
            "code inlines : ", code, "\n",
            "citations    : ", cite))
```

create_article *Create an R Journal article with a modified template for texor.*

Description

Create an R Journal article with a modified template for texor.

Usage

```
create_article(name = "test", edit = TRUE)
```

Arguments

name	the name of the tex file, will default to "test"
edit	Opens the file for editing in RStudio/R GUI.

Details

Outputs an LaTeX R Journal paper template set of files in the project directory.

find_wrapper *find wrapper file*

Description

Finds a different named wrapper file for RJournal article

Usage

```
find_wrapper(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Value

wrapper file name or empty string if none

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::find_wrapper(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

generate_rmd *Modify Markdown to R-markdown*

Description

generate rmarkdown file in output folder

Usage

```
generate_rmd(article_dir, web_dir = TRUE, interactive_mode = FALSE)
```

Arguments

article_dir	path to the directory which contains tex article
web_dir	option to create a new web directory, default TRUE
interactive_mode	interactive mode for converting articles with options. default FALSE

Value

R-markdown file in the web folder

Note

Use pandoc version greater than or equal to 3.1

Examples

```
# Note This is a minimal example to execute this function
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir2"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <-  xfun::normalize_path(paste(your_article_folder,"article",sep="/"))
texor::include_style_file(your_article_path)
rebib::aggregate_bibliography(your_article_path)
data <- texor::handle_figures(your_article_path,
                    texor::get_texfile_name(your_article_path))
texor::patch_code_env(your_article_path) # Step 4
texor::patch_table_env(your_article_path) # Step 5
texor::patch_equations(your_article_path) # Step 5.5
texor::patch_figure_env(your_article_path)
rmarkdown::pandoc_version()
texor::convert_to_markdown(your_article_path)
texor::generate_rmd(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

get_journal_details *get Journal details*

Description

get Journal details

Usage

```
get_journal_details(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Value

journal details in an object

Examples

```
article_dir <- "/home/user/documents/2022-1/2020-36/"
texor::get_journal_details(article_dir)
```

get_md_file_name *get markdown file name*

Description

get markdown file name

Usage

```
get_md_file_name(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Value

markdown file name

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::get_md_file_name(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

`get_texfile_name` *Get the name of the tex file included within wrapper file*

Description

The wrapper file refers to an external tex file which contains the actual document content.

Usage

```
get_texfile_name(article_dir)
```

Arguments

`article_dir`	path to the directory which contains tex article

Value

String name of the tex-file name

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::get_texfile_name(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

get_wrapper_type *Get the name of the wrapper file in the article dir*

Description

This function gets the wrapper file name from the commonly named R-Journal wrapper files.

Usage

```
get_wrapper_type(article_dir, auto_wrapper = FALSE, interactive_mode = FALSE)
```

Arguments

article_dir	path to the directory which contains tex article
auto_wrapper	automatically creates a wrapper if TRUE, else asks user. default value FALSE
interactive_mode	interactive mode for converting articles with options.

Details

Usually the R journal wrapper files are named either 1. RJwrapper.tex 2. RJwrap.tex 3. wrapper.tex

Value

String with name of wrapper file or empty

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::get_wrapper_type(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

handle_figures *handle figures*

Description

handle figures

Usage

```
handle_figures(article_dir, file_name)
```

Arguments

article_dir	path to the directory which contains tex article
file_name	name of the LaTeX file

Value

A block of figure data for better conversion.

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::handle_figures(your_article_path,texor::get_texfile_name(your_article_path))
unlink(your_article_folder,recursive = TRUE)
```

---

include_style_file *Include Style file*

Description

Includes the Metafix.sty style file

Usage

```
include_style_file(article_dir)
```

Arguments

article_dir	path to the directory which contains tex article

Details

This style file helps texor and pandoc to retain metadata and ease the conversion process.

Value

adds Metafix.sty file in the article_dir also includes it in RJwrapper file.

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::include_style_file(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

latex_to_web *latex to web*

Description

automated function for converting a single RJarticle to web

Usage

```
latex_to_web(
  dir,
  log_steps = TRUE,
  example = FALSE,
  auto_wrapper = TRUE,
  temp_mode = TRUE,
  web_dir = FALSE,
  interactive_mode = FALSE,
  autonumber_eq = FALSE,
  compile_rmd_in_temp = !temp_mode,
  kable_tab = FALSE,
  fig_in_r = FALSE
)
```

Arguments

dir	directory path
log_steps	Enable/Disable Logging of conversion steps
example	for examples only by default keep it FALSE.
auto_wrapper	automatically creates a wrapper if TRUE, else asks user. default value TRUE
temp_mode	temp mode will convert the document in a temporary folder and keep the original article untouched. default value = TRUE
web_dir	option to create a new web directory, default FALSE
interactive_mode	interactive mode for converting articles with options. default FALSE
autonumber_eq	whether to autonumber the equations, default is FALSE
compile_rmd_in_temp	This works only with a forked version of rjtools.
kable_tab	converts to kable table instead of markdown tables, default is FALSE
fig_in_r	whether to include figures in R code chunks, default is FALSE Not recommended to use with CRAN or github version of the rjtools package. (default FALSE)

Value

RJweb article document in /web folder

Note

Use pandoc version greater than or equal to 3.1

Do not set example = TRUE param when working with conversions.

example param is set TRUE in example, to conform with CRAN check restrictions.

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::latex_to_web(your_article_path,log_steps = FALSE, example = TRUE, temp_mode =FALSE)
unlink(your_article_folder, recursive = TRUE)
```

---

log_setup *texor log setup*

Description

a wrapper function for logger package to set up log file for logging

Usage

```
log_setup(article_dir, file_name, namespace, idx)
```

Arguments

article_dir	path to the directory which contains tex article
file_name	name of the log file
namespace	namespace of log file
idx	index of log level

Value

NULL but also creates a log file in the article_dir

Examples

```
dir.create(your_article_folder <- file.path(tempdir(), "exampledir"))
example_files <-  system.file("examples/article", package = "texor")
x <- file.copy(from = example_files,to=your_article_folder,recursive = TRUE)
your_article_path <- paste(your_article_folder,"article",sep="/")
 texor::log_setup(your_article_path, "log-file.log", "texor", 2)
unlink(your_article_folder,recursive = TRUE)
```

pandoc_version_check *check texor pandoc compatibility*

Description

texor package requires minimum pandoc version above or equal to 3.1, hence this utility will check for the installation and version status.

Usage

```
pandoc_version_check()
```

Value

TRUE if v >= 3.1, else FALSE

Examples

```
rmarkdown::pandoc_version()

texor::pandoc_version_check()
```

patch_figure_env *patch figure environments*

Description

This function calls the stream editor to change figure* to figure 1. figure*

Usage

```
patch_figure_env(article_dir, with_alg = TRUE)
```

Arguments

article_dir	path to the directory which contains tex article
with_alg	to include algorihtm environment or not

Value

patches figure environments in LaTeX file and also backs up the old file before modification

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::patch_figure_env(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

`patch_table_env` *patch table environment*

Description

function to modify env and commands in TeX using GNU sed

These are due to the pandoc's limitations and ease in conversion.

Usage

```
patch_table_env(article_dir)
```

Arguments

`article_dir`	path to the directory which contains tex article

Details

changes are made to : 1. table* environment to table environment 2. \multicolumn to \multicolumnx \multicolumnx is redefined in Metafix.sty as \renewcommand{\multicolumnx}[3]{\multicolumn{#1}{c}{#3}}

Value

patches table environments in LaTeX file and also backs up the old file before modification

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::patch_table_env(your_article_path)
unlink(your_article_folder,recursive = TRUE)
```

pre_conversion_statistics *pre conversion statistics*

Description

count common environments,inlines for debugging purposes

Usage

```
pre_conversion_statistics(article_dir, write_yaml = TRUE)
```

Arguments

article_dir	path to the directory which contains RJ article
write_yaml	write to a yaml file (default = TRUE)

Value

conversion stat block with details also a yaml file if param enabled.

Examples

```
article_dir <- system.file("examples/article",
                 package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::patch_code_env(your_article_path)
texor::patch_table_env(your_article_path)
texor::patch_equations(your_article_path)
texor::patch_figure_env(your_article_path)
texor::pre_conversion_statistics(your_article_path,write_yaml = FALSE)
unlink(your_article_folder,recursive = TRUE)
```

produce_html *call rmarkdown::render to generate html file*

Description

call rmarkdown::render to generate html file

Usage

```
produce_html(
  article_dir,
  example = FALSE,
  web_dir = TRUE,
  interactive_mode = FALSE
)
```

Arguments

article_dir	path to the directory which contains tex article
example	only enabled for running examples for documentation and to enable export of this function.
web_dir	option to create a new web directory, default TRUE
interactive_mode	interactive mode for converting articles with options. default FALSE

Value

Renders a RJwrapper.html file in the /web folder, in example it will return TRUE

Note

Use pandoc version greater than or equal to 3.1

Do not set example = TRUE param when working with conversions.

example param is set TRUE in example, to conform with CRAN check restrictions.

Examples

```
# Note This is a minimal example to execute this function
article_dir <- system.file("examples/article",
                package = "texor")
dir.create(your_article_folder <- file.path(tempdir(), "tempdir"))
x <- file.copy(from = article_dir, to = your_article_folder,recursive = TRUE,)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::include_style_file(your_article_path)
rmarkdown::pandoc_version()
texor::convert_to_markdown(your_article_path)
texor::generate_rmd(your_article_path)
texor::copy_other_files(your_article_path)
texor::produce_html(your_article_path,example = TRUE)
unlink(your_article_folder,recursive = TRUE)
```

rnw_generate_rmd *Modify Markdown from Sweave to R-markdown*

Description

generate rmarkdown file in output folder

Usage

```
rnw_generate_rmd(
  article_dir,
  web_dir = TRUE,
  interactive_mode = FALSE,
  output_format,
  autonumber_eq = FALSE,
  autonumber_sec = TRUE,
  algorithm_render = FALSE
)
```

Arguments

article_dir	path to the directory which contains tex article
web_dir	option to create a new web directory, default TRUE
interactive_mode	interactive mode for converting articles with options. default FALSE
output_format	knit output type for the RMarkdown file, options for "bookdown", "biocstyle", "litedown"
autonumber_eq	whether to autonumber the equations, default is FALSE
autonumber_sec	whether to autonumber the sections, default is TRUE
algorithm_render	Enable to include algorithms with pseudocode.js, default is FALSE optional is TRUE

Value

R-markdown file in the web folder

Note

Use pandoc version greater than or equal to 3.1

Examples

```
# Note This is a minimal example to execute this function
# Please refer to texor::rnw_to_rmd for a detailed example
```

rnw_to_rmd *Sweave to RMarkdown*

Description

automated function for converting a single Sweave file to R Markdown file

Usage

```
rnw_to_rmd(
  input_file,
  output_format,
  clean_up = TRUE,
  autonumber_eq = FALSE,
  autonumber_sec = TRUE,
  suppress_package_startup_message = FALSE,
  kable_tab = TRUE,
  fig_in_r = TRUE,
  algorithm_render = FALSE
)
```

Arguments

input_file	input Sweave file path
output_format	knit output type for the RMarkdown file options for "bookdown", "biocstyle", "litedown"
clean_up	whether to clean up the intermediate files, default is TRUE
autonumber_eq	whether to autonumber the equations, default is FALSE
autonumber_sec	whether to autonumber the sections, default is TRUE
suppress_package_startup_message	whether to suppress the package startup message, default is FALSE
kable_tab	converts to kable table instead of markdown tables
fig_in_r	whether to include figures in R code chunks, default is TRUE
algorithm_render	Enable to include algorithms with pseudocode.js, default is FALSE optional is TRUE

Value

True if R Markdown file successfully generated in the same folder

Note

Use pandoc version greater than or equal to 3.1

Examples

```
# move example Sweave article and associated files to a temporary directory
example_dir <- system.file("examples", "sweave_article", package = "texor")
file.copy(from = example_dir, to = tempdir(), recursive = TRUE)
article_dir <- file.path(tempdir(), "sweave_article")

# convert example Sweave article to Rmd
rnw_to_rmd(file.path(article_dir, "example.Rnw"),
           output_format = "bookdown",
           clean_up = TRUE,
           autonumber_eq = TRUE,
           autonumber_sec = FALSE)

# convert Rmd to HTML (comment this step to avoid failure on R CMD Check)
# rmarkdown::render(file.path(article_dir, "example.Rmd"))
# browseURL(file.path(article_dir, "example.html"))

# remove temporary files
unlink(article_dir, recursive = TRUE)
```

`stream_editor` *stream editor*

Description

R equivalent of GNU-sed

Usage

```
stream_editor(raw_lines, pattern, target, replacement)
```

Arguments

`raw_lines`	a vector of readLines from the file
`pattern`	a regex pattern to match
`target`	target string to be replaced
`replacement`	replacement string to be substituted

Value

raw_lines : modified vector of lines

Examples

```
example_string <- "\\target{} \\not_a_target{}"
texor::stream_editor(example_string,"\\s*\\\\target\\{\\}", "\\\\target", "\\\\hit")
```

texor_log *log messages for various categories*

Description

a wrapper function for logging different types of log entries

Usage

```
texor_log(message, category, idx)
```

Arguments

message	message to be sent
category	category of the log message
idx	index of log level

Value

NUll, but also appends message to the log file in article_dir

Examples

```
dir.create(your_article_folder <- file.path(tempdir(), "exampledir"))
example_files <- system.file("examples/article", package = "texor")
x <- file.copy(from = example_files,to=your_article_folder,recursive = TRUE)
your_article_path <- paste(your_article_folder,"article",sep="/")
texor::log_setup(your_article_path, "log-file.log", "texor" , 2)
texor::texor_log("Hello", "INFO", 2)
cat(readLines(paste(your_article_path,"/log-file.log",sep="")),sep="\n")
unlink(your_article_folder,recursive = TRUE)
```

Index