

Package ‘excerptr’

October 26, 2025

Title Excerpt Structuring Comments from Your Code File and Set a Table of Contents

Version 2.1.0

Description Ever read or wrote source files containing sectioning comments? If these comments are markdown style section comments, you can excerpt them and set a table of contents using the 'python' package 'excerpts' (<<https://pypi.org/project/excerpts/>>).

URL <https://gitlab.com/fvafrcu/excerptr>

Depends R (>= 3.3.0)

Imports checkmate, reticulate

License BSD_2_clause + file LICENSE

SystemRequirements Python (>= 3.0.0)

Encoding UTF-8

RoxygenNote 7.3.3

Suggests testthat, knitr, rmarkdown, devtools, RUnit, pkgload, fritools, rprojroot

VignetteBuilder knitr

NeedsCompilation no

Author Andreas Dominik Cullmann [aut, cre]

Maintainer Andreas Dominik Cullmann <fvafrcu@mailbox.org>

Repository CRAN

Date/Publication 2025-10-26 20:30:02 UTC

Contents

excerptr-package	2
excerptr	2
Index	4

excerptr-package	<i>Excerpt Structuring Comments from Your Code File and Set a Table of Contents</i>
------------------	---

Description

This is just an R interface to the python package excerpts (<https://pypi.org/project/excerpts/>).

Author(s)

Maintainer: Andreas Dominik Cullmann <fvafrcu@mailbox.org>

See Also

Useful links:

- <https://gitlab.com/fvafrcu/excerptr>

excerptr	<i>Excerpt Structuring Comments and Set a Table of Contents.</i>
----------	--

Description

Just a wrapper to the python function excerpts.excerpts. See <https://pypi.org/project/excerpts/> for details.

Usage

```
excerptr(  
  file_name,  
  comment_character = "#",  
  magic_character = "%",  
  output_path = "",  
  allow_pep8 = TRUE,  
  prefix = "",  
  postfix = "",  
  run_pandoc = TRUE,  
  compile_latex = FALSE,  
  pandoc_formats = "tex"  
)
```

Arguments

<code>file_name</code>	Name the file from which to excerpt.
<code>comment_character</code>	Should be "#" for R.
<code>magic_character</code>	The magic character marking the structuring comments in <code>file_name</code> .
<code>output_path</code>	Name the path where to output to.
<code>allow_pep8</code>	Don't bother. See <code>excerpts.excerpts</code> for details.
<code>prefix</code>	Give a prefix to use for output files.
<code>postfix</code>	Give a postfix to use for output files.
<code>run_pandoc</code>	Try to run pandoc on the output?
<code>compile_latex</code>	Try to compile LaTeX if pandoc has output LaTeX?
<code>pandoc_formats</code>	A character vector naming the output formats for pandoc.

Value

The status of `excerpts.excerpts`.

Examples

```
if (reticulate::py_available(initialize = TRUE)) {  
  root <- system.file(package = "excerptr")  
  test_files <- file.path(root, "excerpts", "tests", "files")  
  outpath <- tempdir()  
  python_source <- "some_code.py"  
  f <- normalizePath(file.path(test_files, python_source))  
  cat(readLines(f), sep = "\n")  
  tryCatch(  
    excerpt(file_name = file.path(test_files, python_source),  
            output_path = tempdir(), run_pandoc = FALSE,  
            compile_latex = FALSE,  
            pandoc_formats = c("tex", "html")),  
    error = identity)  
  f <- file.path(outpath, sub("\\.py$", ".md", python_source))  
  if (!fritools::is_cran()) cat(readLines(f), sep = "\n")  
}
```

Index

* **package**

 excerptr-package, [2](#)

excerptr, [2](#)

excerptr-package, [2](#)