Package 'fpeek'

February 9, 2026

Type Package

Title Check Text Files Content at a Glance

Version 0.2.0

Description Tools to help text files importation. It can return the number of lines; print the first and last lines; convert encoding; guess delimiters and file encoding. Operations are made without reading the entire file before starting, resulting in good performances with large files. This package provides an alternative to a simple use of the 'head', 'tail', 'wc' and 'iconv' programs that are not always available on machine where R is installed.

License MIT + file LICENSE

URL https://github.com/davidgohel/fpeek, https://davidgohel.github.io/fpeek/

BugReports https://github.com/davidgohel/fpeek/issues

Imports Rcpp (>= 0.12.12)

Suggests covr, readr, testthat

LinkingTo Rcpp

Encoding UTF-8

RoxygenNote 7.3.3

NeedsCompilation yes

Author David Gohel [aut, cre]

Maintainer David Gohel <david.gohel@ardata.fr>

Repository CRAN

Date/Publication 2026-02-09 11:10:02 UTC

Contents

`peek_count_lines` *number of lines of a file*

Description

return the number of lines found in a file. Operation is counting the number of new line symbols in the file.

Usage

```
peek_count_lines(path, with_eof = FALSE)
```

Arguments

`path`	file path
`with_eof`	count the end of file as a new line.

Value

number of lines as an integer

Examples

```
f <- system.file(package = "fpeek",
  "datafiles", "test-tab.csv")
peek_count_lines(f)
```

`peek_guess_delim` *Guess Delimited File Parameters*

Description

Guess the delimiter, quote character and decimal mark of a delimited text file. The function splits each of the first `n` lines by each candidate delimiter and selects the delimiter that produces the most consistent number of fields.

The algorithm is adapted from the vroom package.

Usage

```
peek_guess_delim(
  path,
  delims = c(",", "\t", " ", "|", ":", ";"),
  quotes = c("\"", "'"),
  n = 1024
)
```

Arguments

path	path to the text file.
delims	character vector of candidate delimiters.
quotes	character vector of candidate quote characters.
n	number of lines to read for guessing (default 1024).

Value

a named list with elements:

delim the guessed delimiter character (or NULL)

quote the guessed quote character (or NULL)

decimal_mark the guessed decimal mark (or NULL)

Examples

```
f <- system.file(package = "fpeek",
  "datafiles", "test-comma.csv")
peek_guess_delim(f)

f <- system.file(package = "fpeek",
  "datafiles", "test-semicolon.csv")
peek_guess_delim(f)
```

---

peek_guess_encoding *Guess File Encoding*

Description

Detect the encoding of a text file. This function is a wrapper around guess_encoding from the readr package, returning the best candidate as a character string.

readr must be installed (it is listed in Suggests). If it is not available, the function stops with an informative message.

Usage

```
peek_guess_encoding(path)
```

Arguments

path	path to the text file.

Value

a character string giving the most likely encoding.

Examples

```
## Not run:
f <- system.file(package = "fpeek",
  "datafiles", "cigfou-ISO-8859-1.txt")
peek_guess_encoding(f)

## End(Not run)
```

---

peek_head *print the first lines of files*

Description

print the first n lines of a file.

Usage

```
peek_head(path, n = 10, intern = FALSE)
```

Arguments

path	file path
n	number of lines to print
intern	a logical which indicates whether to capture the output as an R character vector or to print the output in the R console.

Examples

```
f <- system.file(package = "fpeek",
  "datafiles", "test-tab.csv")
peek_head(f, n = 4)
peek_head(f, n = 4, intern = TRUE)
```

peek_iconv *Converts encoding of characters*

Description

Read a file, convert the encoding of characters and print the result.

Usage

```
peek_iconv(path, from, to = "UTF-8", newfile = NULL)
```

Arguments

path	file path
from	the code set in which the input is encoded.
to	the code set to which the output is to be converted.
newfile	result file. Default to NULL. If null the result will be print in the R console, otherwise a file is produced containing the result.

Examples

```
la_cigale <- system.file(package = "fpeek", "datafiles",
  "cigfou-ISO-8859-1.txt")

peek_iconv(la_cigale, from = "ISO-8859-1", to = "UTF-8")

newfile <- tempfile()
peek_iconv(la_cigale, from = "ISO-8859-1", to = "UTF-8",
  newfile = newfile)
peek_head(newfile, n = 10)
```

peek_tail *print the last lines of files*

Description

print the last n lines of a file.

Usage

```
peek_tail(path, n = 10, intern = FALSE)
```

Arguments

path	file path
n	number of lines to print
intern	a logical which indicates whether to capture the output as an R character vector or to print the output in the R console.

Examples

```
f <- system.file(package = "fpeek",
  "datafiles", "test-tab.csv")
peek_tail(f, n = 4)
peek_tail(f, n = 4, intern = TRUE)
```

Index