Package ‘fortunes’

October 13, 2022

Version 1.5-4

Date 2016-12-29

Title R Fortunes

Author Achim Zeileis (R code) and the R community (fortunes). Contributions (fortunes and/or code) by Torsten Hothorn, Peter Dalgaard, Uwe Ligges, Kevin Wright, Martin Maechler, Kjetil Brinchmann Halvorsen, Kurt Hornik, Duncan Murdoch, Andy Bunn, Ray Brownrigg, Roger Bivand, Spencer Graves, Jim Lemon, Christian Kleiber, David L. Reiner, Berton Gunter, Roger Koenker, Charles Berry, Marc Schwartz, Michael Dewey, Ben Bolker, Peter Dunn, Sarah Goslee, Simon Blomberg, Bill Venables, Roland Rau, Thomas Petzoldt, Rolf Turner, Mark Leeds, Emmanuel Charpentier, Chris Evans, Paolo Sonego, Peter Ehlers, Detlef Steuer, Tal Galili, Greg Snow, Brian D. Ripley, Michael Sumner, David Winsemius, Liviu Andronic, Brian Diggs, Matthieu Stigler, Michael Friendly, Dirk Eddelbuettel, Richard M. Heiberger, Patrick Burns, Dieter Menne, Andrie de Vries, Barry Rowlingson, Renaud Lancelot, R. Michael Weylandt, Jon Olav Skoien, Francois Morneau, Antony Unwin, Joshua Wiley, Terry Therneau, Bryan Hanson, Henrik Singmann, Eduard Szoecs, Gregor Passolt, John C. Nash.

Maintainer Achim Zeileis <Achim.Zeileis@R-project.org>

Description A collection of fortunes from the R community.

Imports utils

License GPL-2 | GPL-3

NeedsCompilation no

Repository CRAN

Date/Publication 2016-12-29 13:50:47

R topics documented:

fortunes *R Fortunes*

Description

Read and print R fortunes.

Usage

```
fortune(which = NULL, fortunes.data = NULL, fixed = TRUE,
        showMatches = FALSE, author = character(), ...)
## S3 method for class 'fortune'
print(x, width = NULL, ...)
read.fortunes(file = NULL)
```

Arguments

`which` an integer specifying the row number of `fortunes.data`. Alternatively `which` can be a character and `grep` is used to try to find a suitable row.

`fortunes.data` data frame containing a fortune in each row. By default the fortune data from the **fortunes** package are used.

`fixed` logical passed to `grep` if `which` is a character, indicating if it should work (if `TRUE`, as by default) with a simple character string or (if `FALSE`) with regular expressions.

`showMatches` if `which` is character, a logical indicating if `fortune()` should print all the row numbers of `fortunes.data` which match the `grep` search.

`author` a `character` string to match (via `grep`) to the `"authors"` column of `fortunes.data`.

`...` potential further arguments passed to `grep`.

`x` an object of class `"fortune"`, usually a single row from `fortunes.data`.

`width` integer specifying the character width. By default `getOption("width")` is used.

`file` a character string giving a fortune database in csv format (in UTF-8 encoding). By default all csv files in the data directory of the fortune package are used.

Value

`fortune()` returns an object of class `"fortune"` which is a row from a data frame of fortunes (like those read in from `read.fortunes`).

`read.fortunes()` returns a data frame of fortunes, each row contains:

`quote` the quote, main part of the fortune,

`author` the author of the quote,

`context` the context in which it was quoted (if available, otherwise `NA`),

`source` where it was quoted (if available, otherwise `NA`),

`date` when it was quoted (if available, otherwise `NA`).

Examples

```
fortune() # a random one
fortune("Ripley") # a random one from those with 'Ripley'
fortune(author = "Ripley") # a random one from those by 'Ripley'
fortune(17)

fortune("parse", showMatches = TRUE) # -> shows at least 5 matches
fortune("parse.*answer") # nothing found but...
fortune("parse.*answer", fixed = FALSE) # ...this works


## The first three "all together"   ('setNames()' requires at least R 3.0.0):
lapply(setNames(, c(38, 106, 129)), fortune)
```

Index