Package ‘crossword.r’

January 20, 2019

Date 2019-01-20
Type Package
Title Generating Crosswords from Word Lists
Version 0.3.6
Author Peter Meissner
Maintainer Peter Meissner <retep.meissner@gmail.com>
Description Generate crosswords from a list of words.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
Imports R6 (>= 2.2.0), dplyr (>= 0.5.0), stringr (>= 1.2.0), magrittr
(>= 1.5), jsonlite (>= 1.5), r6extended (>= 0.1.1)
RoxygenNote 6.0.1
Suggests covr, testthat
NeedsCompilation no
Repository CRAN
Date/Publication 2019-01-20 06:40:03 UTC

R topics documented:

Crossword .......................................................... 2
cw_greplv ......................................................... 3
cw_matrix_to_df .................................................. 3
cw_normalize_words ............................................ 4
cw_to_json ......................................................... 4
cw_wordlist_animal_en .......................................... 4
%>% ............................................................... 5

Index 6
Crossword

Description
Crossword

Usage
Crossword

Format
An `R6Class` generator object for generating crosswords from word lists

Fields
letters a character matrix representing the grid of the crossword
words a data.frame like (tibble) storing words, their position on the grid (row, col), their length in character, their direction ("right", "down") the word and the clue

Methods
add_words(words, clues = NULL) this method will try to add words to the crossword by placing it on the grid; clues is optional and should be the same length;
density() gives back statistics on fill state of grid
to_json(pretty = FALSE) this exports grid and word list data to JSON for external usage; pretty parameter determines if this is done in a human readable or more machine efficient way

Examples
library(crossword)
cw <- Crossword$new(rows = 4, columns = 4)
cw$add_words(c("back", "nasa", "kick", "nuk", "ic", "sic"))
cw
cw$letters
cw$words
cw$density()
cw_greplv

**Description**

A vectorized version of grep

**Usage**

```
cw_greplv(pattern, x, ignore.case = FALSE, perl = FALSE, fixed = FALSE, useBytes = FALSE)
```

**Arguments**

- `pattern`: character string containing a regular expression (or character string for `fixed = TRUE`) to be matched in the given character vector. Coerced by `as.character` to a character string if possible. If a character vector of length 2 or more is supplied, the first element is used with a warning. Missing values are allowed except for `regexpr` and `gregexpr`.
- `x`: a character vector where matches are sought, or an object which can be coerced by `as.character` to a character vector. Long vectors are supported.
- `ignore.case`: if `FALSE`, the pattern matching is case sensitive and if `TRUE`, case is ignored during matching.
- `perl`: logical. Should Perl-compatible regexps be used?
- `fixed`: logical. If `TRUE`, `pattern` is a string to be matched as is. Overrides all conflicting arguments.
- `useBytes`: logical. If `TRUE` the matching is done byte-by-byte rather than character-by-character. See ‘Details’.

---

cw_matrix_to_df

**Description**

Function that turn matrix into a data.frame in long format

**Usage**

```
cw_matrix_to_df(x)
```

**Arguments**

- `x`: the data.frame to transform
cw_normalize_words (words)

**Arguments**

- **words**
  character vector of words to normalize for crossword usage

---

cw_to_json (cw, pretty = FALSE)

**Arguments**

- **cw**
  an object of class crossword
- **pretty**
  should json formatted to be more human readable or not

---

cw_wordlist_animal_en

**Format**

An object of class data.frame with 68 rows and 2 columns.

**Description**

data frame of words and clues

**Usage**

cw_wordlist_animal_en
Description

re-export magrittr pipe operator
Index

*Topic datasets
  Crossword, 2
  cw_wordlist_animal_en, 4
  as.character, 3

Crossword, 2
  cw_greplv, 3
  cw_matrix_to_df, 3
  cw_normalize_words, 4
  cw_to_json, 4
  cw_wordlist_animal_en, 4

Long vectors, 3

R6Class, 2
  regular expression, 3