

Package ‘tfse’

November 29, 2018

Version 0.4.4

Title Tools for Script Editing

Description A collection of useful tools for programming and writing-scripts. Several functions are simple wrappers around base R functions that extend their functionality while also providing some convenient properties—regular expression functions that automatically detect look-ahead and look-behind statements, a read-line function that suppresses incomplete-final-line warnings and automatically opens and closes connections, a version of substrings that starts from the end of strings, etc. Other functions are useful for working with hexadecimal colors, installing packages, omitting missing data, and showing in-use connections.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0.9000

Depends R (>= 2.10)

Imports grDevices, utils, crayon

Suggests ggplot2, knitr, rmarkdown, testthat, covr

URL <https://tfse.mikewk.com>

BugReports <https://github.com/mkearney/tfse/issues>

NeedsCompilation no

Author Michael W. Kearney [aut, cre] (<<https://orcid.org/0000-0002-0730-4694>>)

Maintainer Michael W. Kearney <kearneymw@missouri.edu>

Repository CRAN

Date/Publication 2018-11-28 23:00:03 UTC

R topics documented:

add_arg_if	3
boxcode	3

cc	4
col2hex	5
count_na	5
data_set	6
desc_get_var	6
emj	7
github_raw	7
gregexpr_	8
install_if	8
match_arg	9
max_na	9
menuline	10
na_omit	10
nin	11
n_uq	12
pbcopy	12
peel_lists	13
pmsg	13
print_as_col	14
print_complete	14
print_start	15
psub	15
readlines	16
readline_	16
regmatches_	17
rename_git_repo	17
rescale	18
rm_DS_Store	19
search_files	19
search_function	20
set_class	20
shhh	21
show_connections	21
substrev	22
tfse	22
theme_mwk	23
trim_ws	23
un_zip	24

add_arg_if	<i>Add defaults to argument list</i>
------------	--------------------------------------

Description

Adds parameters to argument list if list does not already include those parameters

Usage

```
add_arg_if(args, ..., override = FALSE)
```

Arguments

args	Argument list
...	Other named arguments are added (depending on override) and returned with args
override	Logical indicating whether to override existing values in args with the values provided as a named argument here.

Value

Argument list with updated values.

Examples

```
## arg list
args <- list(x = 5, y = TRUE, z = FALSE)

## add arg defaults
add_arg_if(args, w = TRUE, z = TRUE)

## add arg defaults, overriding any previous values
add_arg_if(args, x = 10, z = TRUE, override = TRUE)
```

boxcode	<i>boxcode</i>
---------	----------------

Description

Load clipboard with code chunk square. Paste to insert square into R script file at cursor location.

Usage

```
boxcode(...)
```

```
codebox(...)
```

```
box_code(...)
```

```
code_box(...)
```

Arguments

```
...          Name of section/block
```

Value

Text for code box saved into clipboard. Paste to use at cursor.

```
cc          Combine comma separated strings
```

Description

Split strings by comma into character vector(s)

Usage

```
cc(x, simplify)
```

Arguments

```
x          Vector of comma separated character strings
simplify   Logical indicating whether to return a character vector (the default) when the
           length of x is one. This argument does nothing if the length of x is greater than
           1.
```

Value

If length of x is 1 then a character vector otherwise a list of character vectors.

Examples

```
## comma separated alphabet
abcs <- paste(letters, collapse = ",")

## split single string
cc(abcs)
```

```
## return as list
cc(abcs, simplify = FALSE)

## select columns
mtcars[, cc("cyl,mpg,wt,gear")]

## character vector with multiple strings
x <- c("v1,v2,v3", "y1,y2,y5")

## convert strings into list of [split] character vectors
cc(x)
```

col2hex

col2hex

Description

Converts named colors to hex (html) codes.

Usage

```
col2hex(color, alpha = NULL)
```

Arguments

color	One or more named colors. Also accepts hex colors (useful for adding alpha values).
alpha	Alpha level ranging form 1.0 (opaque) to 0.0 (transparent). The default (NULL) omits alpha columns from the returned hexadecimal characters.

Value

Hexadecimal color codes

count_na

Count missing values

Description

Returns counts of missing values

Usage

```
count_na(x)
```

Arguments

x Input data.

Value

Counts of missing observations

data_set *Create data set*

Description

Simple way to make data frames

Usage

data_set(...)

Arguments

... Data to be converted to data frame

Value

A tibble data frame

desc_get_var *Description variables*

Description

Get values from package DESCRIPTION

Usage

desc_get_var(pkg, field = NULL)

desc_get_url(pkg)

desc_gh_repo(pkg)

Arguments

pkg Character string, name of package

field Character string, name of field/key

Value

Data frame

emj	<i>Emojis for creating print messages</i>
-----	---

Description

A dataset containing emojis useful for message printing.

Usage

```
emj
```

Format

A data frame with 1 row and 30 variables:

github_raw	<i>Generate link to raw Github file</i>
------------	---

Description

Converts Github path/repo/file information into a link to the raw version of the file

Usage

```
github_raw(file, repo = NULL)
```

Arguments

file	Name of desired file; if repo is NULL, then this value should also provide repo information, i.e., owner and name of the repository—e.g., "owner/repo/file.ext". Alternatively, user may supply the URL to the file as it appears in a web browser, e.g., "https://github.com/mkearney/driven-snow/blob/master/theme/driven-snow.rstheme"
repo	Repository name information formatted as username/repo. If this information is provided in the value supplied to file then leave this as NULL (the default)

Value

Returns the URL path to the raw version of the file.

gregexpr_	<i>smart gregexpr wrapper</i>
-----------	-------------------------------

Description

smart gregexpr wrapper

Usage

```
gregexpr_(x, pat, ...)
```

```
regexpr_(x, pat, ...)
```

Arguments

x	Input text
pat	Reg ex pattern
...	Other args passed to base (g)regexpr

Value

Pattern match positions

install_if	<i>Install packages if not already</i>
------------	--

Description

Wrapper around install.packages

Usage

```
install_if(pkg)
```

Arguments

pkg	Character; one or more packages to install.
-----	---

Value

Returns data frame with pkg (chr) and required_install (lgl) columns.

match_arg	<i>match arg to choices</i>
-----------	-----------------------------

Description

Wrapper around [match.arg](#) that defaults to ignoring case and trimming white space

Usage

```
match_arg(arg, choices, multiple = FALSE, ignore_case = TRUE,
          trim_ws = TRUE)
```

Arguments

arg	a character vector (of length one unless several.ok is TRUE) or NULL
choices	a character vector of candidate values
multiple	logical specifying if arg should be allowed to have more than one element. Defaults to FALSE
ignore_case	logical indicating whether to ignore capitalization. Defaults to TRUE
trim_ws	logical indicating whether to trim surrounding white space. Defaults to TRUE

Value

Value(s) matched via partial matching.

max_na	<i>Filter based on proportion of missing data</i>
--------	---

Description

Returns columns that have less than or equal to a specified amount of missingness.

Usage

```
max_na(x, max = 0.05)
```

Arguments

x	Input data frame or matrix.
max	Maximum proportion of missingnesses allowed. Columns with higher proportions of missingness compared to this value will be dropped. Columns will only be returned if they have 1 - max proportion non-missing. This value must be between 0-1. It defaults to .05.

Value

Data frame or matrix with columns with less than or equal to the max allowed proportion of missingness.

menuline	<i>menuline</i>
----------	-----------------

Description

Creates interactive multiple choice question.

Usage

```
menuline(q, a)
```

Arguments

q	Question to be asked in interactive session.
a	Answer choices.

Value

Selection provided by user in interactive session.

na_omit	<i>Omit missing values/rows</i>
---------	---------------------------------

Description

Returns data object with NA values (if atomic vector), all NA elements (if list), or all NA rows (if data frame or matrix) omitted

Usage

```
na_omit(x)
```

Arguments

x	Data object
---	-------------

Value

Data with NA values (if atomic vector), all NA elements (if list), or all NA rows (if data frame or matrix) omitted.

Examples

```
## generate data
df <- data.frame(
  a = I(list(c(1, 2), c(NA_integer_, NA_integer_), c(1, 2))),
  b = c("a", NA_character_, "c"),
  c = c(1.1243, NA_real_, -1.234134)
)

## data frame
na_omit(df)

## matrix
na_omit(as.matrix(df))

## list
na_omit(apply(df, 1, c))

## atomic vector
na_omit(df$b)
```

nin

Return lhs values not in rhs values

Description

Return lhs values not in rhs values

Usage

```
nin(lhs, rhs, value = TRUE)
```

```
yin(lhs, rhs, value = TRUE)
```

Arguments

lhs	Values to check whether they are/not contained in the other
rhs	Values to use as the reference
value	Logical indicating whether to return the value or a logical vector

Examples

```
## a, b, zz in alphabet letters
yin(c("a", "b", "zz"), letters)

## a, b, zz NOT in alphabet letters
nin(c("a", "b", "zz"), letters)
```

n_uq *Number of unique elements*

Description

Estimates number of unique elements in a data object

Usage

n_uq(x)

Arguments

x Input data

Value

Integer number of unique elements

pbcopy *pbcopy*

Description

Adds input to clipboard for pasting

Usage

pbcopy(x)

Arguments

x Input passed to cat function.

Value

Prints x to clipboard.

Examples

```
## alphabet as string
pbcopy(paste(letters, collapse = ""))
## paste e.g., C-v
```

peel_lists	<i>peel lists</i>
------------	-------------------

Description

peel lists

Usage

peel_lists(x)

Arguments

x Input data

Value

Peeled object

pmsg	<i>Paste collapse input and print as message</i>
------	--

Description

Paste collapse input and print as message

Usage

pmsg(..., print = TRUE)

Arguments

...	Strings to be paste0 collapse = ""
print	Logical indicating whether to print the message (default) or return an unevaluated expression

Value

Either invisibly returns text of message or unevaluated expression

print_as_col	<i>Print data frame as a single column</i>
--------------	--

Description

Prints first row of data frame

Usage

```
print_as_col(x)
```

Arguments

x	Input data frame
---	------------------

Value

Prints first row and variable names

print_complete	<i>Print message about completing a task</i>
----------------	--

Description

Prints a check-mark bulleted message presumably about task completion

Usage

```
print_complete(...)
```

Arguments

...	Strings collapsed (with no additional space added) into black message prefixed with a heavy-check emoji (color and emoji print may appear differently depending on your system/UI configuration)
-----	--

Value

A printed message

<code>print_start</code>	<i>Print message about starting a task</i>
--------------------------	--

Description

Prints a next-arrow bulleted message presumably about task completion

Usage

```
print_start(...)
```

Arguments

<code>...</code>	Strings collapsed (with no additional space added) into gray message prefixed with a next-arrow emoji (color and emoji print may appear differently depending on your system/UI configuration)
------------------	--

Value

A printed message

<code>psub</code>	<i>Paste sub</i>
-------------------	------------------

Description

Glue-like sub pasting of strings

Usage

```
psub(x, ...)
```

Arguments

<code>x</code>	Input string
<code>...</code>	Named strings with names being the values to replace and the strings being the desired new value.

readlines	<i>readlines</i>
-----------	------------------

Description

Read lines of file

Usage

```
readlines(x, ...)
```

Arguments

x	Input
...	Other args passed to readLines.

Details

Simple wrapper around [readLines](#) that automates opening and closing of connection file.

Value

Output

readline_	<i>readline_</i>
-----------	------------------

Description

Worry free way to read lines from interactive sessions.

Usage

```
readline_(...)
```

Arguments

...	Character string or vector to be used as prompt during interactive R session. Ultimately, this function only sends a single string to the user, but it will accept a vector if you're picky about not creating strings of a certain width.
-----	--

Value

Input entered during interactive session without extra quotes.

regmatches_	<i>extract reg expr matches</i>
-------------	---------------------------------

Description

A wrapper around the base function combo of gregexpr and regmatches

Usage

```
regmatches_(x, pat, drop = FALSE, ...)
```

```
regmatches_first(x, pat, drop = FALSE, ...)
```

Arguments

x	Text data.
pat	Reg ex pattern
drop	Logical indicating whether to drop empty matches. Defaults to FALSE.
...	Other args (like ignore.case) passed to gregexpr

Value

Matching expression from text.

rename_git_repo	<i>Rename git repo code</i>
-----------------	-----------------------------

Description

Command line (bash) syntax for renaming a git repo

Usage

```
rename_git_repo(new_url = NULL)
```

Arguments

new_url	Optional, URL pointing to correct (renamed) repo. If NULL (default), "new_url" is printed in brackets.
---------	--

Value

text of git command to rename git repo

rescale

Standard: Rescale values to a standard normal scale

Description

Standard: Rescale values to a standard normal scale

Normal: Rescale values to a standard (0-1) scale

Log: Rescale values to a natural log scale

Point-scale: Rescale values to a new point scale

Usage

```
rescale_standard(x, na_omit = TRUE)
```

```
rescale_normal(x, na_omit = TRUE)
```

```
rescale_log(x, na_omit = TRUE)
```

```
rescale_pointscale(x, lower, upper, lower0 = NULL, upper0 = NULL,  
  na_omit = TRUE)
```

Arguments

x	Input vector
na_omit	Logical indicating whether to drop missing (NA) values. Default is TRUE.
lower	Min value of new scale. Only applicable for pointscales.
upper	Max value of new scale. Only applicable for pointscales.
lower0	Min value of old scale. If NULL, defaults to min of input. Only applicable for pointscales.
upper0	Max value of old scale. If NULL, defaults to max of input Only applicable for pointscales.

Value

Rescaled vector

Examples

```
## randomly sample 10 values ranging from -10 to 100  
x <- sample(-10:100, 10)  
  
## rescale to 0-1 scale  
rescale_standard(x)  
  
## rescale to normal distribution (z-scores)  
rescale_normal(x)
```

```
## rescale to logged distribution (natural log)
rescale_log(x)

## rescale to new point scale
rescale_pointscale(x, 1, 7, lower0 = -10, upper0 = 100)
```

rm_.DS_Store	<i>Remove pesky .DS_Store files</i>
--------------	-------------------------------------

Description

Recursively removes all .DS_Store files in working directory.

Usage

```
rm_.DS_Store()
```

Examples

```
## Not run:
rm_.DS_Store()

## End(Not run)
```

search_files	<i>search_files</i>
--------------	---------------------

Description

Returns matching files and line numbers of given string pattern.

Usage

```
search_files(x, path = ".", recursive = TRUE, all.files = FALSE)
```

Arguments

x	Pattern.
path	Path on which to restrict search. Defaults to current working directory.
recursive	logical
all.files	default false excludes dot files

Value

Output from terminal - file name, line number, and preview of matching text

search_function	<i>search functions</i>
-----------------	-------------------------

Description

Like search_files but for functions

Usage

```
search_function(pat, fun)
```

Arguments

pat	Pattern to match
fun	Function to inspect

set_class	<i>set class</i>
-----------	------------------

Description

Set class with a parenthetical function.
Add class with a parenthetical function.

Usage

```
set_class(x, value)  
add_class(x, value)
```

Arguments

x	Object to assign new class to.
value	Class value to assign to x

Value

Object x as class value.

shhh	<i>execute expression quietly</i>
------	-----------------------------------

Description

execute expression quietly

Usage

shhh(expr)

Arguments

expr Expression to be evaluated without additional printing.

Value

Output from evaluated expression.

show_connections	<i>show connections</i>
------------------	-------------------------

Description

Displays active connections as a tidy tibble

Usage

show_connections()

Value

Prints and invisibly returns data frame.

substrev	<i>Sub string reverse</i>
----------	---------------------------

Description

Returns portion of string starting from end of string (otherwise just like substr)

Usage

```
substrev(x, start, stop = 0)
```

Arguments

x	Character vector
start	Number of characters to include relative to the last character position.
stop	Specify the number of characters from the final character to set as the last character position.

Value

Sub string with last i characters.

tfse	<i>tfse: Various Useful Functions</i>
------	---------------------------------------

Description

Collection of useful functions.

Author(s)

Michael W. Kearney

theme_mwk	<i>theme_mwk</i>
-----------	------------------

Description

My ggplot2 theme

Usage

```
theme_mwk(base_size = 12, base_family = "Roboto Condensed",  
          dark = "#24292e", light = "#ffffff", gray = "#ededed",  
          caption = TRUE)
```

```
theme_mwk_caption_text()
```

Arguments

base_size	Base font size
base_family	Font family.
dark	Primary forefront color.
light	Primary background color.
gray	Gray color.
caption	Text to use as source caption.

Value

My ggplot2 theme (similar to theme_minimal/theme_bw)

trim_ws	<i>trim_ws</i>
---------	----------------

Description

Returns character vector without extra spaces and trimmed of white space.

Usage

```
trim_ws(x)
```

Arguments

x	Character vector
---	------------------

Value

Character vector without extra spaces

un_zip	<i>Unzip files into directory</i>
--------	-----------------------------------

Description

Unzip archive into similarly named directory

Usage

un_zip(path)

Arguments

path Name of zipfile. Must end in ".zip"

Value

Creates a directory in the same folder with the same name (minus the zip part)

Index

- *Topic **datasets**
 - emj, [7](#)
- *Topic **package**
 - tfse, [22](#)
- *Topic **tfse-package**
 - tfse, [22](#)

- add_arg_if, [3](#)
- add_class (set_class), [20](#)

- box_code (boxcode), [3](#)
- boxcode, [3](#)

- cc, [4](#)
- code_box (boxcode), [3](#)
- codebox (boxcode), [3](#)
- col2hex, [5](#)
- count_na, [5](#)

- data_set, [6](#)
- desc_get_url (desc_get_var), [6](#)
- desc_get_var, [6](#)
- desc_gh_repo (desc_get_var), [6](#)

- emj, [7](#)

- github_raw, [7](#)
- gregexpr_, [8](#)

- install_if, [8](#)

- match.arg, [9](#)
- match_arg, [9](#)
- max_na, [9](#)
- menuline, [10](#)

- n_uq, [12](#)
- na_omit, [10](#)
- nin, [11](#)

- pbcopy, [12](#)

- peel_lists, [13](#)
- pmsg, [13](#)
- print_as_col, [14](#)
- print_complete, [14](#)
- print_start, [15](#)
- psub, [15](#)

- readline_, [16](#)
- readLines, [16](#)
- readlines, [16](#)
- regexpr_ (gregexpr_), [8](#)
- regmatches_, [17](#)
- regmatches_first (regmatches_), [17](#)
- rename_git_repo, [17](#)
- rescale, [18](#)
- rescale_log (rescale), [18](#)
- rescale_normal (rescale), [18](#)
- rescale_pointscale (rescale), [18](#)
- rescale_standard (rescale), [18](#)
- rm_.DS_Store, [19](#)

- search_files, [19](#)
- search_function, [20](#)
- set_class, [20](#)
- shhh, [21](#)
- show_connections, [21](#)
- substrev, [22](#)

- tfse, [22](#)
- tfse-package (tfse), [22](#)
- theme_mwk, [23](#)
- theme_mwk_caption_text (theme_mwk), [23](#)
- trim_ws, [23](#)

- un_zip, [24](#)

- yin (nin), [11](#)