Package ‘rprintf’

Type Package
Title Adaptive Builder for Formatted Strings
Version 0.2.1
Author Kun Ren <ken@renkun.me>
Maintainer Kun Ren <ken@renkun.me>
Description Provides a set of functions to facilitate building formatted strings under various replacement rules: C-style formatting, variable-based formatting, and number-based formatting. C-style formatting is basically identical to built-in function ’sprintf’. Variable-based formatting allows users to put variable names in a formatted string which will be replaced by variable values. Number-based formatting allows users to use index numbers to represent the corresponding argument value to appear in the string.
Depends R (>= 2.15)
Date 2015-09-09
Imports stringi
Suggests testthat, knitr
License MIT + file LICENSE
URL http://renkun.me/rprintf, https://github.com/renkun-ken/rprintf
BugReports https://github.com/renkun-ken/rprintf/issues
ByteCompile TRUE
NeedsCompilation no
Repository CRAN
Date/Publication 2015-09-09 09:34:46

R topics documented:

rprintf ......................................................... 2
rprintr ......................................................... 3
rprintv ......................................................... 3

Index 5
rprintf  

Build a character vector or list with adaptive string formatting

Description

The `rprintf` function checks the given character vector or list and applies appropriate formatters that transform it from generic patterns to specific texts with variables and indices as placeholders replaced by a given set of values in correct formats.

Usage

`rprintf(x, ...)`

Arguments

- `x` The character vector or list to be transformed
- `...` The arguments that specify the set of values to be placed

Examples

```r
## Not run:
#' # Format a single-entry character vector with sprintf mechanism
rprintf('Hello, %s','world')
rprintf('%s (%d years old)','Ken',24)
rprintf('He is %d but has a height of %.1fcm',18,190)

# Format a single-entry character vector with variable mechanism
rprintf('Hello, $name', name = 'world')
rprintf('He is $age but has a height of $height:.2f cm',age=18,height=190)
rprintf('$a, $b:.2f, $c:.2f', a=1.56,b=2.34,c=3.78)

# Format a single-entry character vector with numbering mechanism
rprintf('Hello, {1}', 'world')
rprintf('He is {1} but has a height of {2:.2f} cm',18,190)
rprintf('{{1},{2:.1f}, {3:.2f}, {2}, {1:.0f}},{1.56,2.34,3.78}
rprintf('{{2},{1}},{x},{y})

# This function also works for character vectors and lists.
rprintf(c('%s:%d','$name:$age','(1):(2)'),name = 'Ken',age = 24)
rprintf(c(a = '%s:%d',b = '$name:$age',c = '(1):(2)'),name = 'Ken',age = 24)
rprintf(list('%s:%d','$name:$age','(1):(2)'),name = 'Ken',age = 24)
rprintf(list(a = '%s:%d',b = '$name:$age',c = '(1):(2)'),name = 'Ken',age = 24)

# It also works with list argument for named variables.
p <- list(name = 'Ken',age = 24)
rprintf('name: $name, age: $age',p)
rprintf('name: {1}, age: {2}',p)
```
Note that when the list of arguments are given names, the variable names in format string should be modified.

```r
rprintf('name: $arg.name, age: $arg.age', arg = p)
```

## End(Not run)

---

### rprintf

**Build a character vector or list with number-based string formatting**

**Description**

The `rprintf` function applies number-based formatter to transform the given character vector to specific texts with numbers replaced by a given set of values in correct formats.

**Usage**

```r
rprintf(x, ...)```

**Arguments**

- `x` The character vector or list to be transformed
- `...` The arguments that specify the set of values to be placed

**Examples**

```r
## Not run:

# Format a single-entry character vector with numbering mechanism
rprintf('Hello, {1}', 'world')
rprintf('{1} is {2} years old', 'Ken', 24)
rprintf('He is {1} but has a height of {2:.2f}cm', 18, 190)
rprintf('{1}, {2:.1f}, {3:+.2f}, {2}, {1:.0f}', 1.56, 2.34, 3.78)
rprintf('{2},{1}', 'x', 'y')
```

## End(Not run)

---

### rprintv

**Build a character vector or list with variable-based string formatting**

**Description**

The `rprintv` function applies variable-based formatter to transform the given character vector to specific texts with named variables replaced by a given set of values in correct formats.
Usage

\texttt{rprintv(x, \ldots)}

Arguments

\texttt{x} \hspace{1cm} The character vector or list to be transformed
\texttt{\ldots} \hspace{1cm} The arguments that specify the set of values to be placed

Examples

\texttt{## Not run:}

\texttt{
# Format a single-entry character vector with variable mechanism
rprintf('Hello, \$name', name='world')
rprintf('\$name ($age years old'),name='Ken',age=24)
rprintf('He is $age but has a height of $height:.2fcm',age=18,height=190)
rprintf('$a, $b:.1f, $c:.2f, $b, $a:.0f',a=1.56,b=2.34,c=3.78)

## End(Not run)
Index

rprintf, 2
rprintrn, 3
rprintv, 3