

Package ‘commafree’

April 19, 2024

Title Call Functions Without Commas Between Arguments

Version 0.2.0

Description Provides the “comma-free call” operator: ‘%(%)’. Use it to call a function without commas between the arguments. Just replace the ‘(’ with ‘%(%)’ in a function call, supply your arguments as standard R expressions enclosed by ‘{ }’, and be free of commas (for that call).

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.1

URL <https://github.com/t-kalinowski/commafree>

BugReports <https://github.com/t-kalinowski/commafree/issues>

NeedsCompilation no

Author Tomasz Kalinowski [aut, cre]

Maintainer Tomasz Kalinowski <kalinowskit@gmail.com>

Repository CRAN

Date/Publication 2024-04-19 13:12:43 UTC

R topics documented:

%(%) 1

Index 4

%(%) *Call a function*

Description

This allows you to call a function with expressions for arguments. It is especially useful for long, multi-line function calls with many arguments (e.g., a shiny UI definition, an R6 class definition, ...)

Usage

```
fn %% args
```

Arguments

fn	A function
args	A set of expressions grouped by { }

Details

This %%() merely performs a syntax transformation, so all the same semantics with regards to lazy argument evaluation apply. For any function call, replace (with %% and be free of the need for commas between arguments in that call.

```
fn %% {
  a
  b
  c
}
```

Is syntactically equivalent to writing:

```
func(
  a,
  b,
  c
)
```

Value

Whatever fn() called with args returns.

Note

You can produce a missing argument with the special token ,, or foo = ` , ` for a named missing arguments (see examples).

Examples

```
mean %% {
  1:3
  na.rm = TRUE
}
```

```
writeLines(c %% {
  "Hello"
  "Goodbye"
})
```

```
# setup helper demonstrating missing arguments
fn <- function(x, y) {
  if(missing(y))
    print("y was missing")
  else
    print(y)
}

# How to add a named missing argument
fn %% {
  y = ``,`
}

# How to add a positional missing argument
fn %% {
  1
  ``,`
}

fn %% { 1; ``,` }

rm(fn) # cleanup
```

Index

%(%, 1