

Package: arg (via r-universe)

May 6, 2026

Title A simple modern interface to command line arguments

Version 0.0.0.9000

Description This package provides a wrapper around {optparse} to easily handle command line arguments.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports optparse, tibble, std

Repository <https://rogiersbart.r-universe.dev>

Date/Publication 2025-06-10 12:56:46 UTC

RemoteUrl <https://github.com/rogiersbart/arg>

RemoteRef HEAD

RemoteSha dc2c7c7ff26cc4dd4c408d2ee0cfc11efd883bff

Contents

c	1
hello	2
v	2

Index	4
--------------	----------

c	<i>Argument count</i>
---	-----------------------

Description

Access the number of arguments in the argument vector.

Usage

```
c()
```

Value

The number of raw arguments provided in the command line call.

Examples

```
assignInNamespace("arguments", base::c("one", "two", "three"), "arg") # for interactive testing
arg::v()
```

hello	<i>Hello, World!</i>
-------	----------------------

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

v	<i>Argument vector</i>
---	------------------------

Description

Access the argument vector. If an argument(s) and/or option(s) specification is provided, the argument vector is preprocessed first, and a named list with arguments and options is returned instead.

Usage

```
v(..., title = "Documentation", header = "", footer = "", expose = FALSE)
```

Arguments

...	A <code>tibble::tribble()</code> argument(s) and/or option(s) specification, with name, short, long, default, type and help columns.
title	The title for the documentation.
header	An introductory header for the documentation.
footer	A closing footer for the documentation.
expose	Should the arguments and options be attached to the calling environment instead of being returned as a named list?

Value

A named list of processed options and arguments, except if ... is not defined, in case which the raw argument vector is returned, or if expose is TRUE, in case which the options and arguments are attached to the calling environment and nothing is returned.

Examples

```
assignInNamespace("arguments", base::c("one", "2", "-o", "-h"), "arg") # for interactive testing
arg::v(
  ~name, ~short, ~long, ~default, ~type, ~help,
  "arg1", NA, NA, "default_arg1_value", "character", "First argument",
  "arg2", NA, NA, "1", "integer", "Second argument",
  "opt1", "o", "option1", NA, "logical", "Logical option",
  "opt2", "p", "option2", "Default", "character", "Character option",
  header = "This command line tool provides functionality to ..."
)
```

Index

c, 1

hello, 2

v, 2