

Package ‘Rapp’

January 20, 2025

Title Easily Build Command Line Applications

Version 0.2.0

Description Run simple 'R' scripts as command line applications, with automatic robust and convenient support for command line arguments. This package provides 'Rapp', an alternative 'R' front-end similar to 'Rscript', that enables this.

License MIT + file LICENSE

URL <https://github.com/r-lib/Rapp>

BugReports <https://github.com/r-lib/Rapp/issues>

Encoding UTF-8

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Imports yaml

NeedsCompilation no

Author Tomasz Kalinowski [aut, cre]

Maintainer Tomasz Kalinowski <tomasz@posit.co>

Repository CRAN

Date/Publication 2024-07-15 17:10:02 UTC

Contents

run	2
Index	4

run	<i>Run an R app.</i>
-----	----------------------

Description

Run an R app.

Usage

```
run(app, args = commandArgs(TRUE))
```

Arguments

app	A filepath to an Rapp.
args	character vector of command line args.

Details

See the package README for full details. <https://github.com/r-lib/Rapp>

Value

NULL, invisibly. Called for its side effect.

Examples

```
# For the example, place 'Rapp', the package examples, and 'R' on the PATH
old_path <- Sys.getenv("PATH")
Sys.setenv(PATH = paste(system.file("examples", package = "Rapp"),
                        system.file("exec", package = "Rapp"),
                        R.home("bin"),
                        old_path,
                        sep = .Platform$path.sep))

# Here is an example app:
# flip-coin.R
writeLines(readLines(
  system.file("examples/flip-coin.R", package = "Rapp")))

if(.Platform$OS.type != "windows") {
  # on macOS and Linux, you can call the app directly
  system("flip-coin.R")
  system("flip-coin.R --n 3")
} else {
  # On windows, there is no support for shebang '#!' style executables
  # but you can invoke 'Rapp' directly
  system("Rapp flip-coin.R")
  system("Rapp flip-coin.R --n 3")
}
```

run

3

```
# restore PATH  
Sys.setenv(PATH = old_path)
```

Index

run, [2](#)