

Package ‘funr’

July 22, 2025

Title Simple Utility Providing Terminal Access to all R Functions

Description A small utility which wraps Rscript and provides access to all R functions from the shell.

Version 0.3.2

License MIT + file LICENSE

LazyData true

Imports utils

URL <https://github.com/sahilseth/funr>

BugReports <https://github.com/sahilseth/funr/issues>

RoxygenNote 5.0.1

Suggests testthat

NeedsCompilation no

Author Sahil Seth [aut, cre]

Maintainer Sahil Seth <me@sahilseth.com>

Repository CRAN

Date/Publication 2016-04-19 08:12:03

Contents

funr	2
get_script_path	3
render_funr	3
setup	4
sys.script	4
Index	5

funr*funr: providing a simple command-line interface to R functions*

Description

Wraps Rscript in a easy to use manner, exposing all R functions from the terminal. The [github page](#) provides more details with examples, highlights and caveats.

Usage

```
funr(args, help_text, script_name = "funr")
```

Arguments

args	Should always be: <code>commandArgs(trailingOnly = TRUE)</code> , when used inside a script. Example
help_text	A simple text to be displayed describing options and usage of this interface. Supplying this, replaces the default text.
script_name	Name of the script. This is used in the the help text. [funr]

Source

<https://github.com/sahilseth/funr>

Examples

```
## show funr help
## terminal version: funr -h
funr()

## show rnorm help
## terminal version: funr -h rnorm
render_funr(funr(args=c("-h", "rnorm")))

## Generate a few random numbers
## terminal version: funr rnorm n=10
render_funr(funr(args=c("rnorm", "n=10")))
```

get_script_path	<i>Enables detection of the folder a script resides in with certain accuracy</i>
-----------------	--

Description

R does not have a default way to return and use, the location of a specific script.

Usage

```
get_script_path()
```

```
detect_home()
```

Source

https://github.com/molgenis/molgenis-pipelines/wiki/How-to-source-another_file.R-from-within-your-R-script

render_funr	<i>Render output of functions in a print friendly format</i>
-------------	--

Description

If the function returns with invisible, output is suppressed

Usage

```
render_funr(x, max_rows = 100)
```

Arguments

x	a output from funr
max_rows	In case output of a function is a data.frame, the number of rows to display.

setup	<i>Setup and initialize some scripts.</i>
-------	---

Description

Setup and initialize some scripts.

Usage

```
setup(bin = "~/bin")
```

Arguments

bin	path to bin folder
-----	--------------------

Details

Will add more to this to identify cluster and aid in other things

sys.script	<i>Get path of current script</i>
------------	-----------------------------------

Description

Get path of current script

Usage

```
sys.script()
```

Source

<http://stackoverflow.com/questions/1815606/rscript-determine-path-of-the-executing-script>

Examples

```
## Not run:  
## cat foo.R  
## library(funr);sys.script()  
  
## source("foo.R")  
  
## Rscript foo.R  
  
## End(Not run)
```

Index

`cli (funr)`, [2](#)

`detect_home (get_script_path)`, [3](#)

`funr`, [2](#)

`get_script_path`, [3](#)

`render_funr`, [3](#)

`rfun (funr)`, [2](#)

`setup`, [4](#)

`sys.script`, [4](#)