

Package ‘curlconverter’

August 29, 2016

Title Tools to Transform 'cURL' Command-Line Calls to 'httr' Requests

Version 0.7.0

Maintainer Bob Rudis <bob@rudis.net>

Description Deciphering web/'REST' 'API' and 'XHR' calls can be tricky, which is one reason why internet browsers provide ``Copy as cURL" functionality within their ``Developer Tools" pane(s). These 'cURL' command-lines can be difficult to wrangle into an 'httr' 'GET' or 'POST' request, but you can now ``straighten" these 'cURLs' either from data copied to the system clipboard or by passing in a vector of 'cURL' command-lines and getting back a list of parameter elements which can be used to form 'httr' requests. You can also make a complete/working/callable 'httr::VERB' function right from the tools provided.

SystemRequirements xclip <<http://sourceforge.net/projects/xclip/>> or xsel <<http://www.vergenet.net/~conrad/software/xsel/>> for accessing the X11 clipboard

Depends R (>= 3.0.0)

Imports V8, curl, httr, purrr, clipr, stringi, formatR, magrittr, jsonlite

License AGPL

LazyData true

Suggests testthat, knitr, rmarkdown

RoxygenNote 5.0.1

VignetteBuilder knitr

NeedsCompilation no

Author Bob Rudis [aut, cre],
Nick Carneiro [ctb]

Repository CRAN

Date/Publication 2016-03-29 20:04:36

R topics documented:

curlconverter	2
make_req	2

parse_query	3
straighten	4
%>%	5

Index	6
--------------	----------

curlconverter	<i>Create 'httr' requests from "Copy as cURL" command-line calls</i>
---------------	--

Description

Tools to Transform 'cURL' Command-line Calls to 'httr' Requests

Details

Deciphering web/'REST' 'API' and 'XHR' calls can be tricky, which is one reason why internet browsers provide "Copy as cURL" functionality within their "Developer Tools" pane(s). These 'cURL' command-lines can be difficult to wrangle into an 'httr' 'GET' or 'POST' request, but you can now "straighten" these 'cURLs' either from data copied to the system clipboard or by passing in a vector of 'cURL' command-lines and getting back a list of parameter elements which can be used to form 'httr' requests. These lists can be passed to another function to automatically make 'httr' functions.

Author(s)

Bob Rudis (@hrbrmstr)

make_req	<i>Turn parsed cURL command lines into httr request functions</i>
----------	---

Description

Takes the output of `straighten()` and turns the parsed cURL command lines into working `httr` `VERB()` functions, optionally cat'ing the text of each function to the console and/or replacing the system clipboard with the source code for the function.

Usage

```
make_req(x, use_parts = FALSE, quiet = TRUE, add_clip = (length(x) == 1))
```

Arguments

x	a vector of curlcoverter objects
use_parts	logical. If TRUE, the request function will be generated from the "URL parts" that are created as a result of the call to straighten . This is useful if you want to modify the URL parts before calling make_req. Default: FALSE.
quiet	if FALSE, will cause make_req() to write complete function source code to the console.
add_clip	if TRUE, will overwrite the system clipboard with the character string contents of the last newly made 'httr::VERB' function (i.e. this is intended to be used in a workflow where only one cURL command line is being processed). Defaults to TRUE if length(x) is 1

Value

a list of working R functions.

References

[Evaluating Network Performance, Network Monitor](#)

See Also

[straighten\(\)](#), [httr VERB\(\)](#)

Examples

```
library(httr)

my_ip <- straighten("curl 'https://httpbin.org/ip'") %>% make_req()

## Not run:
# external test which captures live data
content(my_ip[[1]](), as="parsed")

## End(Not run)
```

parse_query

Split a query string into component parts

Description

While [straighten](#) will returns parsed GET query string parameters there are times (i.e. in HTML <form> processing) when the body of the request contains a URL encoded query string as well.

Usage

```
parse_query(query)
```

Arguments

query query string to decode

Details

This function will take any query string and return a named list of the parameters. Both the names and values will be URL decoded.

References

[Query Strings](#)

Examples

```
parse_query("a=1&b=test")
```

straighten

Processes cURL command-line requests

Description

Takes in a *"Copy as cURL"* command line and returns a list of components that can be used to build httr requests or passed to `make_req()` to automatically make an httr `VERB()` function.

Usage

```
straighten(curl$ = read_clip(), quiet = FALSE)
```

Arguments

curl\$ a character vector of one or more cURL command lines. It will read from the clipboard (i.e. if you did a *"Copy as cURL"* from browser developer tools).

quiet if FALSE, a message with the original cURL command line will be output. (Default: FALSE)

Value

list of length(curl\$) containing parsed data (i.e. to be used in 'httr' requests)

References

[Evaluating Network Performance, Network Monitor](#)

See Also

`make_req()`, `httr VERB()`

Examples

```
library(httr)

my_ip <- straighten("curl 'https://httpbin.org/ip'") %>% make_req()

## Not run:
# external test which captures live data
content(my_ip[[1]](), as="parsed")

## End(Not run)
```

%>%

Pipe operator

Description

Pipe operator

Index

`%>%`, [5](#)

`curlconverter`, [2](#)

`curlconverter-package (curlconverter)`, [2](#)

`make_req`, [2](#), [4](#)

`parse_query`, [3](#)

`straighten`, [2](#), [3](#), [4](#)

`VERB`, [2-4](#)