

# Package ‘postlightmercury’

October 14, 2022

**Type** Package

**Title** Parses Web Pages using Postlight Mercury

**Version** 1.2

**Author** Mikkel Freltoft Krogsholm

**Maintainer** Mikkel Freltoft Krogsholm <mikkel@56n.dk>

**Description** This is a wrapper for the Mercury Parser API. The Mercury Parser is a single API endpoint that takes a URL and gives you back the content reliably and easily.

With just one API request, Mercury takes any web article and returns only the relevant content — headline, author, body text, relevant images and more — free from any clutter. It’s reliable, easy-to-use and free.

See the webpage here: <<https://mercury.postlight.com/>>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**Imports** tibble, crul, purrr, jsonlite, rvest, xml2

**Suggests** testthat, covr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-07-09 06:26:56 UTC

## R topics documented:

null_to_na . . . . .	2
remove_html . . . . .	2
web_parser . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

null_to_na	<i>Turns NULL values in a list into NAs.</i>
------------	----------------------------------------------

---

**Description**

Turns NULL values in a list into NAs.

**Usage**

```
null_to_na(mylist)
```

**Arguments**

`mylist` is a list, where the NULL values are to be turned into NAs.

---

remove_html	<i>Removes html</i>
-------------	---------------------

---

**Description**

The function uses tools from the rvest and xml2 packages to clean up the HTML and turning it into proper text.

**Usage**

```
remove_html(strings, trim = TRUE)
```

**Arguments**

`strings` the string(s) you want to clean  
`trim` should the string be trimmed or not

**Value**

a string

**Examples**

```
## Not run:
# First get api key here: https://mercury.postlight.com/web-parser/

# Then run the code below replacing the X's wih your api key.
url <- "https://trackchanges.postlight.com/building-awesome-cms-f034344d8ed"
my_data <- web_parser(page_urls = url,
                      api_key = XXXXXXXXXXXXXXXXXXXXXXXX)

# With html formatting:
```

```
my_data$content

# Now remove it:
my_data$content <- remove_html(my_data$content)

# Without html formatting:
my_data$content

## End(Not run)
```

---

web_parser	<i>Parses web pages</i>
------------	-------------------------

---

## Description

With just one API request, Mercury takes any web article and returns only the relevant content — headline, author, body text, relevant images and more — free from any clutter. It's reliable, easy-to-use and free.

## Usage

```
web_parser(page_urls, api_key)
```

## Arguments

page_urls	One or more urls to be parsed
api_key	Key for the API

## Value

a tibble

## Source

<https://mercury.postlight.com/web-parser/>

## Examples

```
## Not run:
# First get api key here: https://mercury.postlight.com/web-parser/

# Then run the code below replacing the X's with your api key:
web_parser(page_urls = "https://trackchanges.postlight.com/building-awesome-cms-f034344d8ed",
           api_key = XXXXXXXXXXXXXXXXXXXXXXXX)

## End(Not run)
```

# Index

`null_to_na`, 2

`remove_html`, 2

`web_parser`, 3