

# Package: carbonate (via r-universe)

September 21, 2024

**Title** Interact with 'carbon.js'

**Version** 0.1.4

**Description** Create beautiful images of source code using  
'carbon.js' <<https://carbon.now.sh/about>>.

**License** MIT + file LICENSE

**URL** <https://github.com/yonicd/carbonate>

**BugReports** <https://github.com/yonicd/carbonate/issues>

**Depends** R (>= 3.2.0)

**Imports** clipr, details, httpuv, httr, magick, R6, RSelenium, rtweet,  
utils, wdmn, jsonlite, yaml

**Suggests** covr, knitr, rmarkdown, testthat

**VignetteBuilder** knitr

**RdMacros** details

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE, old\_usage = TRUE, r6 = FALSE)

**RoxygenNote** 7.2.1

**Repository** <https://yonicd.r-universe.dev>

**RemoteUrl** <https://github.com/yonicd/carbonate>

**RemoteRef** HEAD

**RemoteSha** 071a9f828e40c90c69925489ca2813fb128be206

## Contents

.browse	2
.carbonate	3
.chromeOptions	4
.encode	4
.firefoxOptions	5
.get_windows_control_themes	5

<a href="#">.options</a>	6
<a href="#">.rtweet</a>	6
<a href="#">.set_template</a>	7
<a href="#">.start</a>	8
<a href="#">.tiny</a>	9
<a href="#">.uri</a>	9
<a href="#">carbon</a>	10
<a href="#">carbon_fields</a>	11
<a href="#">tinyurl</a>	13

<b>Index</b>	<b>14</b>
--------------	-----------

---

<a href="#">.browse</a>	<i>open \$uri to in browser window</i>
-------------------------	--

---

## Description

opens the uri address in [\\$uri](#) in the browser window.

## Usage

```
.browse(self, private)
```

## Arguments

self	carbon self object
private	carbon private object

## See Also

[carbon](#)

## Examples

```
x <- carbon$new('x <- 1')
if(interactive())
  x$browse()
```

---

.carbonate

*Carbonate script lines to a carbon image*

---

### Description

Main function of the package that invokes RSelenium to open a browser to the carbon.js uri, create an image and download the file.

### Usage

```
.carbonate(self, private, file, path, code, rD, driver)
```

### Arguments

self	carbon self object
private	carbon private object
file	character, name of file to save image as
path	character, path to save the image
code	character, lines of script to make carbon image from
rD	RSelenium driver
driver	character, select either 'firefox' or 'chrome' driver

### Details

Script is passed to <https://carbon.now.sh/> is downloaded to the `tempdir()` and appended to the list `$carbons` using RSelenium and Chrome.

### Value

image object

### See Also

[carbon](#), [rsDriver](#)

### Examples

```
if(interactive()){  
  x <- carbon$new('x <- 1')  
  x$carbonate()  
}
```

---

`.chromeOptions`      *chrome webdriver functions*

---

**Description**

Functions to open,close,set chrome webdriver

**Usage**

```
.chromeOptions(self, private)
```

**Arguments**

<code>self</code>	carbon self object
<code>private</code>	carbon private object

**See Also**

[carbon](#), [chrome](#)

---

`.encode`      *encode uri for carbon.js*

---

**Description**

encode uri string for carbon.js. Augmented version of than [URLencode](#)

**Usage**

```
.encode(self, private, URL, reserved, repeated)
```

**Arguments**

<code>self</code>	carbon self object
<code>private</code>	carbon private object
<code>URL</code>	character, uri string to url encode
<code>reserved</code>	boolean, should 'reserved' characters be encoded?
<code>repeated</code>	boolean, should apparently already-encoded URLs be encoded again?

**Value**

character

**See Also**

[URLencode](#), [carbon](#)

---

.firefoxOptions      *gecko webdriver functions*

---

### **Description**

Functions to open,close,set gecko webdriver

### **Usage**

```
.firefoxOptions(self, private)
```

### **Arguments**

self	carbon self object
private	carbon private object

### **See Also**

[carbon](#), [gecko](#)

---

.get\_windows\_control\_themes  
*carbon get functions*

---

### **Description**

function that gets the available list of aesthetic values of fields

### **Usage**

```
.get_windows_control_themes(self, private)  
  
.get_font_families(self, private)  
  
.get_templates(self, private)
```

### **Arguments**

self	carbon self object
private	carbon private object

### **See Also**

[carbon](#)

---

`.options`                      *concatenate the carbon options to a string*

---

### Description

combine all the carbon options into a carbon.js valid string

### Usage

```
.options(self, private, code)
```

### Arguments

<code>self</code>	carbon self object
<code>private</code>	carbon private object
<code>code</code>	character, script to embed into the uri

### Value

OUTPUT\_DESCRIPTION

### See Also

[carbon](#)

---

`.rtweet`                      *send tweets using carbonate outputs*

---

### Description

wrapper for [post\\_tweet](#) to send tweets with media created using carbonate.

### Usage

```
.rtweet(self, private, media, status = self$tweet_status,
  media_format = c("png", "gif"), ...)
```

### Arguments

<code>self</code>	carbon self object
<code>private</code>	carbon private object
<code>media</code>	magick-image object or path to image
<code>status</code>	character, status to attach to post, Default: <code>self\$tweet_status</code>
<code>media_format</code>	character, type of media to tweet, Default: <code>c('png','gif')</code>
<code>...</code>	arguments to pass to <a href="#">post_tweet</a>

## Details

If multiple images are passed they can be either converted to a gif by setting `media_format` to 'gif' or attached each one to the post.

## Value

outcome from rtweet

## See Also

[image\\_write\\_post\\_tweet](#)

## Examples

```
## Not run:
x <- carbonate::carbon$new()
x$carbonate(code = readLines(system.file('DESCRIPTION',package='carbonate')))
x$carbonate(code = 'x+2')

# using default status value (x$tweet_status)
x$rtweet(system.file('figures/hex_black_small.png',package='carbonate'))

x$rtweet(status = 'these are two pngs',media = x$carbons,media_format='png')
x$rtweet(status = 'this is a gif', media = x$carbons,media_format='gif')

## End(Not run)
```

---

.set\_template

*carbon set functions*

---

## Description

function that set the aesthetics of the carbon image output

## Usage

```
.set_template(self, private, template)

.set_window_control_theme(self, private, theme)

.set_font_family(self, private, family)
```

**Arguments**

self	carbon self object
private	carbon private object
template	character, name of template to set <a href="#">\$template</a> to
theme	character, theme to set <a href="#">\$theme</a> to
family	character, font family to set <a href="#">\$font_family</a> to

**See Also**

[carbon](#)

---

.start *RSelenium webdriver functions*

---

**Description**

Functions to open, stop, stop\_all RSelenium sessions

**Usage**

```
.start(self, private, eCap, driver = c("firefox", "chrome"))
.driver_start(self, private, driver = c("chrome", "firefox"))
.driver_stop(self, private)
.stop(self, private)
.stop_all(self, private)
.set_port(self, private, port = NULL)
.get_port(self, private)
```

**Arguments**

self	carbon self object
private	carbon private object
eCap	chromeOptions passed into <a href="#">rsDriver</a>
driver	character, select either 'firefox' or 'chrome' driver
port	integer, port for the <a href="#">rsDriver</a> to use, if NULL then a random port is selected

**See Also**

[carbon](#)



---

.tiny                      *convert carbon uri to tinyurl*

---

**Description**

convert carbon uri to tinyurl with option to attach to clipboard.

**Usage**

```
.tiny(self, private, clip = FALSE)
```

**Arguments**

self	carbon self object
private	carbon private object
clip	boolean, write tinyurl to clipboard, Default: FALSE

**Details**

If clip is set to TRUE [write\\_clip](#) will put the tinyurl on the clipboard.

**Value**

character

---

.uri                      *URI constructor*

---

**Description**

Function that is called in a carbon object to construct valid uri to send to browser

**Usage**

```
.uri(self, private, code)
```

**Arguments**

self	carbon self object
private	carbon private object
code	character, script to embed into the uri

**See Also**

[carbon](#)

**Examples**

```

if(interactive()){

x <- carbon$new('x <- 1')

# populate from self$code
x$uri()

# enter manually
x$uri(code = 'x <- y + 3')

}

```

---

carbon

*Carbon R6 class*


---

**Description**

The Carbon generator creates a new Carbon-object, which is the class containing all the app logic. The class is based on the [R6](#) OO-system and is thus reference-based with methods and data attached to each object, in contrast to the more well known S3 and S4 systems.

**Initialization**

A new 'Carbon'-object is initialized using the new() method on the generator:

```
x <- carbon$new(code = clipr::read_clip())
```

**Fields**

Description of fields of the R6 object that can be set by the user can be found in the following [page](#).

**Methods**

[\\$carbonate](#) Using RSelenium fetch the carbon image output  
[\\$browse](#) open [\\$uri](#) in the browser

[\\$set\\_template](#) set \$template  
[\\$get\\_templates](#) get a list of possible templates  
[\\$set\\_window\\_control\\_theme](#) set \$windows\_control\_theme  
[\\$get\\_windows\\_control\\_themes](#) get a list of possible window control themes  
[\\$set\\_font\\_family](#) set \$font\_family  
[\\$get\\_font\\_families](#) get a list of possible fonts

[\\$uri](#) construct valid carbon.js uri

**\$options** return all current carbon options  
**\$encode** URL encode a string for the \$uri  
**\$tiny** Create tinyurl from \$uri  
**\$rtweet** Send tweets containing media generated from carbon

**\$chromeOptions** construct a chromeOptions object  
**\$chrome\_start** start a chrome session  
**\$chrome\_stop** stop a chrome session  
**\$start** start a RSelenium session  
**\$stop** stop a RSelenium session  
**\$stop\_all** stop all active RSelenium sessions  
**\$get\_port** Get active port  
**\$set\_port** Set new port

---

 carbon\_fields

*Carbon object fields*


---

## Description

Fields that can be changed in the [Carbon](#) object

## Arguments

**code** character, script to embed, Default: NULL  
**palette** numeric, layout palletete, Default: c(r=171,g=184,b=195,a=1)  
**template** character, layout template, Default: 'seti'  
**add\_window\_control**  
     boolean, add/remove window controls (circles on top left corner), Default: TRUE  
**window\_control\_theme**  
     character, change window control themes, Default: 'none'  
**language** character, language, Default: 'r'  
**add\_drop\_shadow**  
     boolean, add/remove dropshadow, Default: TRUE  
**drop\_shadow\_offset\_y**  
     numeric, shadow vertical offset (px), Default: 20  
**drop\_shadow\_blur\_radius**  
     numeric, shadow blur radius (px), Default: 68  
**auto\_adjust\_width**  
     boolean, auto-audjust output width, Default: TRUE  
**padding\_vertical**  
     numeric, vertical padding (px), Default: 48

padding_horizontal	numeric, horizontal padding (px), Default: 32
add_line_number	boolean, add/remove line numbers, Default: FALSE
font_family	character, layout font family, Default: 'Hack'
font_size	numeric, font size (px), Default: 14
line_height_percent	numeric, relative space between lines (percent), Default: 133
square_image	boolean, output image is square, Default: FALSE
relative_export_size	numeric, image size in export relative to what is in the preview (1,2 or 4), Default: 1
add_watermark	boolean, add official carbon.js watermark, Default: FALSE
add_tinyurl	boolean, add tinyurl link to image directing to the carbon page that created the image, Default: FALSE
tinyurl_location	character, string with <a href="#">gravity</a> value to place the annotate tinyurl address on image, Default: 'southeast'
tweet	boolean, tweet out the image using <a href="#">post_tweet</a>
tweet_status	character, status to add to the tweet. If add_tinyurl is TRUE then a link to the url will be appended to the status automatically, Default: 'Created in R using the Carbonate (Package Icon)'
add_timestamp	boolean, add timestamp to output filename, Default: FALSE
carbons	list, storage of rendered carbon images, Default: NULL
download_path	character, path to download the images, Default: tempdir()
chrome_args	character, arguments passed to chrome browser, Default: c('--disable-gpu', '--window-size=1280,800')
chrome_pref	list, preferences passed to chrome browser, Default: list("profile.default_content_settings.popups" = 0L, "download.prompt_for_download" = FALSE, "download.default_directory" = tempdir())
rD	RSelenium Driver object
cDrv	Chrome Driver object from wdman

**See Also**[carbon](#)

---

<code>tinyurl</code>	<i>convert uri to tinyurl</i>
----------------------	-------------------------------

---

**Description**

convert uri to tinyurl.

**Usage**

```
tinyurl(uri)
```

**Arguments**

`uri` character, the uri to convert

**Value**

character

# Index

`.browse`, 2  
`.carbonate`, 3  
`.chromeOptions`, 4  
`.driver_start(.start)`, 8  
`.driver_stop(.start)`, 8  
`.encode`, 4  
`.firefoxOptions`, 5  
`.get_font_families`  
    (`.get_windows_control_themes`),  
    5  
`.get_port(.start)`, 8  
`.get_templates`  
    (`.get_windows_control_themes`),  
    5  
`.get_windows_control_themes`, 5  
`.options`, 6  
`.rtweet`, 6  
`.set_font_family(.set_template)`, 7  
`.set_port(.start)`, 8  
`.set_template`, 7  
`.set_window_control_theme`  
    (`.set_template`), 7  
`.start`, 8  
`.stop(.start)`, 8  
`.stop_all(.start)`, 8  
`.tiny`, 9  
`.uri`, 9  
`$browse`, 10  
`$carbonate`, 10  
`$carbons`, 3  
`$chromeOptions`, 11  
`$chrome_start`, 11  
`$chrome_stop`, 11  
`$encode`, 11  
`$font_family`, 8  
`$get_font_families`, 10  
`$get_port`, 11  
`$get_templates`, 10  
`$get_windows_control_themes`, 10  
`$options`, 11  
`$rtweet`, 11  
`$set_font_family`, 10  
`$set_port`, 11  
`$set_template`, 10  
`$set_window_control_theme`, 10  
`$start`, 11  
`$stop`, 11  
`$stop_all`, 11  
`$template`, 8  
`$theme`, 8  
`$tiny`, 11  
`$uri`, 2, 10, 11  
  
Carbon, 11  
carbon, 2–6, 8, 9, 10, 12  
carbon-browse (`.browse`), 2  
carbon-carbonate (`.carbonate`), 3  
carbon-chrome (`.chromeOptions`), 4  
carbon-encode (`.encode`), 4  
carbon-fields (`carbon_fields`), 11  
carbon-gecko (`.firefoxOptions`), 5  
carbon-get-fields  
    (`.get_windows_control_themes`),  
    5  
carbon-options (`.options`), 6  
carbon-rtweet (`.rtweet`), 6  
carbon-selenium (`.start`), 8  
carbon-set-fields (`.set_template`), 7  
carbon-tinyurl (`.tiny`), 9  
carbon-uri (`.uri`), 9  
carbon\_fields, 11  
carbonate (`.carbonate`), 3  
chrome, 4  
  
gecko, 5  
gravity, 12  
  
image\_write, 7  
page, 10

post\_tweet, [6](#), [7](#), [12](#)

R6, [10](#)

rsDriver, [3](#), [8](#)

tinyurl, [13](#)

URLencode, [4](#)

write\_clip, [9](#)