

Package ‘vembedr’

July 16, 2017

Title Functions to Embed Video in HTML

Version 0.1.3

Description A set of functions for generating HTML to embed hosted video in your R Markdown documents or Shiny applications.

Depends R (>= 3.3.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports htmltools, httr, stringr, magrittr

RoxygenNote 6.0.1.9000

URL <https://github.com/ijlyttle/vembedr>

BugReports <https://github.com/ijlyttle/vembedr/issues>

Suggests testthat, knitr, rmarkdown, covr

VignetteBuilder knitr

NeedsCompilation no

Author Ian Lyttle [aut, cre, cph],
Schneider Electric [cph]

Maintainer Ian Lyttle <ian.lyttle@schneider-electric.com>

Repository CRAN

Date/Publication 2017-07-16 18:49:29 UTC

R topics documented:

build_suggestion	2
embed	3
embed_url	4
hms	5
rickroll	5
secs	6

use_start_time	6
vembedr	8
%>%	8

Index	9
--------------	----------

build_suggestion	<i>Given a parse-list, generate an embed-list</i>
------------------	---

Description

This is an internal function, supporting [suggest_embed](#)

Usage

```
build_suggestion(parse_list)
```

Arguments

parse_list, list generated using [parse_video_url](#) with members:
service character, describes which service is used
id character, identifier for the video at the service
start_time character, indicates start time

Value

list with members:

embed character, code for [embed](#) call

start_time character, (optional) code for [use_start_time](#) call

See Also

[suggest_embed](#) [parse_video_url](#)

Examples

```
parse_video_url("https://youtu.be/1-vcEr0PofQ?t=28s") %>%
  build_suggestion()
```

embed

Embed a video into an HTML document

Description

These functions are used to embed video into your **rmarkdown** html-documents, or into your **shiny** apps. There are functions to embed from YouTube, Vimeo, and Microsoft Channel 9 (who host the UseR! 2016 videos).

Usage

```
embed_vimeo(id, width = 500, height = 281, frameborder = 0,  
  allowfullscreen = TRUE, query = NULL, fragment = NULL)
```

```
embed_youtube(id, width = 420, height = 315, frameborder = 0,  
  allowfullscreen = TRUE, query = NULL)
```

```
embed_user2016(id, width = 560, height = 315, frameborder = 0,  
  allowfullscreen = TRUE)
```

```
embed_user2017(id, width = 560, height = 315, frameborder = 0,  
  allowfullscreen = TRUE)
```

```
embed_channel9(id, width = 560, height = 315, frameborder = 0,  
  allowfullscreen = TRUE)
```

Arguments

id	character, identifier provided by the service
width	numeric, width of iframe (px)
height	numeric, height of iframe (px)
frameborder	numeric, size of frame border (px)
allowfullscreen	logical, indicates if to allow fullscreen
query	list of items to include in url-query string
fragment	character, string to include as url-fragment

Details

These services allow you to customize a lot of things by specifying an optional query string. The specification for the query string will differ according to the service being used:

YouTube https://developers.google.com/youtube/player_parameters

Vimeo <https://developer.vimeo.com/player/embedding>

Channel 9 <https://channel9.msdn.com/Events/user-international-R-User-conference/user2016/Forty-years-of-S> (click the *Embed* tab)

Value

An embed object that prints an `htmltools::tags$iframe` element

See Also

[use_start_time](#)

Examples

```
embed_youtube("dQw4w9WgXcQ")
embed_vimeo("45196609")
embed_user2016("Literate-Programming")
embed_user2017("Room-202-Lightning-Talks") %>% use_start_time("26m35s")
```

embed_url

Embed a video based on URL

Description

You can use this function to embed video using only the URL and you do not need any customization beyond the start-time. It should work for all the services supported by the [embed](#) family of functions.

Usage

```
embed_url(url)
```

Arguments

url character, URL of webpage for video

Details

This function calls [suggest_embed](#) then parses and evaluates the code. If you need to customize the iframe, [suggest_embed](#) may be more useful to you.

Value

An embed object that prints an `htmltools::tags$iframe` element

See Also

[suggest_embed](#)

Examples

```
embed_url("https://youtu.be/1-vcErOPofQ?t=28s")
```

hms *Create an hours-minutes-seconds string*

Description

Create an hours-minutes-seconds string

Usage

```
hms(x)
```

Arguments

`x`, numeric (number of seconds), or character (i.e. "3m15s")

Value

character string (i.e. "0h3m15s")

See Also

[secs](#)

rickroll *Embed a popular video*

Description

If you want to experiment with the arguments to [embed](#), such as `query`, but do not have a particular video in mind, this function may be useful to you.

Usage

```
rickroll_vimeo(...)
```

```
rickroll_youtube(...)
```

```
rickroll_channel9(...)
```

Arguments

`...` arguments (other than `id`) passed on to [embed](#)

Value

An embed object that prints an `htmltools::tags$iframe` element

Examples

```
rickroll_vimeo()  
rickroll_youtube()  
rickroll_channel19()
```

secs	<i>Get number of seconds given a string</i>
------	---

Description

This is a helper function to get the number of seconds.

Usage

```
secs(x)
```

Arguments

x character, describes a time duration, i.e. "3m15s"

Details

This could be useful for composing query parameters for YouTube embeds.

Value

numeric, number of seconds

See Also

[embed_youtube](#), [hms](#)

use_start_time	<i>Specify a start time for an embedded video</i>
----------------	---

Description

This function provides you a consistent way to specify the start time, regardless of the service.

Usage

```
use_start_time(...)  
  
## Default S3 method:  
use_start_time(...)  
  
## S3 method for class 'embed_youtube'  
use_start_time(embed, start_time, ...)  
  
## S3 method for class 'embed_vimeo'  
use_start_time(embed, start_time, ...)  
  
## S3 method for class 'embed_channel9'  
use_start_time(embed, start_time, is_paused = TRUE,  
  ...)
```

Arguments

...	generic arguments to pass through
embed	embed object made using an embed function
start_time	numeric (seconds), or character ("3m15s")
is_paused	logical, for "Channel 9" specifies if the video should be paused at this time

Details

The `start_time` argument can take a variety of formats; these inputs all evaluate to the same value:

- "0h1m0s", "0h01m00s", "0h1m"
- "1m0s", "1m"
- "60s", 60

Please note that for Vimeo, you can specify a start time, but you can not specify that the video be paused at this time. In other words, it is like "autoplay" is set to TRUE, and you cannot unset it.

Examples

```
rickroll_youtube() %>%  
  use_start_time("3m32s")
```

 vembedr

vembedr: A package for embedding video

Description

The vembedr package provides two categories of functions: `embed` and `rickroll`, to embed videos from the services YouTube, Vimeo, and Microsoft Channel 9 (including UseR! 2016 and 2017 videos).

Details

The important functions are `embed_vimeo`, `embed_youtube`, and `embed_user2016`, with a helper function for specifying the start times of videos: `use_start_time`.

 %>%

Pipe functions

Description

Like dplyr, vembedr also uses the pipe function, %>% to turn function composition into a series of imperative statements.

Arguments

`lhs`, `rhs` An embed object and a function to apply to it

Examples

```
# Instead of
use_start_time(rickroll_youtube(), "1m35s")
# you can write
rickroll_youtube() %>% use_start_time("1m35s")
```


Index

[%>%](#), 8

[build_suggestion](#), 2

[embed](#), 2, 3, 4, 5, 7

[embed_channel9](#) ([embed](#)), 3

[embed_url](#), 4

[embed_user2016](#), 8

[embed_user2016](#) ([embed](#)), 3

[embed_user2017](#) ([embed](#)), 3

[embed_vimeo](#), 8

[embed_vimeo](#) ([embed](#)), 3

[embed_youtube](#), 6, 8

[embed_youtube](#) ([embed](#)), 3

[hms](#), 5, 6

[parse_video_url](#), 2

[rickroll](#), 5

[rickroll_channel9](#) ([rickroll](#)), 5

[rickroll_vimeo](#) ([rickroll](#)), 5

[rickroll_youtube](#) ([rickroll](#)), 5

[secs](#), 5, 6

[suggest_embed](#), 2, 4

[tags](#), 4, 5

[use_start_time](#), 2, 4, 6, 8

[vembedr](#), 8

[vembedr-package](#) ([vembedr](#)), 8