Package ‘vembedr’

October 10, 2020

<table>
<thead>
<tr>
<th>Title</th>
<th>Embed Video in HTML</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>0.1.4</td>
</tr>
<tr>
<td>Description</td>
<td>A set of functions for generating HTML to embed hosted video in your R Markdown documents or Shiny applications.</td>
</tr>
<tr>
<td>Depends</td>
<td>R (&gt;= 3.3.0)</td>
</tr>
<tr>
<td>License</td>
<td>MIT + file LICENSE</td>
</tr>
<tr>
<td>Encoding</td>
<td>UTF-8</td>
</tr>
<tr>
<td>LazyData</td>
<td>true</td>
</tr>
<tr>
<td>Imports</td>
<td>htmltools, rlang, httr, stringr, magrittr, glue, assertthat, utils</td>
</tr>
<tr>
<td>RoxygenNote</td>
<td>7.1.1</td>
</tr>
<tr>
<td>URL</td>
<td><a href="https://github.com/ijlyttle/vembedr">https://github.com/ijlyttle/vembedr</a></td>
</tr>
<tr>
<td>BugReports</td>
<td><a href="https://github.com/ijlyttle/vembedr/issues">https://github.com/ijlyttle/vembedr/issues</a></td>
</tr>
<tr>
<td>Suggests</td>
<td>testthat, knitr, rmarkdown, covr, here, fs, conflicted, usethis</td>
</tr>
<tr>
<td>VignetteBuilder</td>
<td>knitr</td>
</tr>
<tr>
<td>NeedsCompilation</td>
<td>no</td>
</tr>
<tr>
<td>Author</td>
<td>Ian Lyttle [aut, cre] (<a href="https://orcid.org/0000-0001-9962-4849">https://orcid.org/0000-0001-9962-4849</a>), Schneider Electric [cph]</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Ian Lyttle <a href="mailto:ian.lyttle@se.com">ian.lyttle@se.com</a></td>
</tr>
<tr>
<td>Repository</td>
<td>CRAN</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2020-10-10 21:20:02 UTC</td>
</tr>
</tbody>
</table>

R topics documented:

- vembedr-package ........................................... 2
- embed ...................................................... 2
- embed_url .................................................. 5
- get_service ............................................... 6
vembedr-package  

**vembedr**: Package for embedding video

## Description

The vembedr package lets you embed video into your HTML pages for these services:

## Details

- YouTube
- Vimeo
- Microsoft Channel 9
- Box
- Microsoft Stream

It provides two categories of functions:

- **embed** functions, to specify a video to embed: e.g. `embed_youtube()`, `embed_url()`
- **use** functions, to modify the embedding: e.g. `use_start_time()`, `use_rounded()`

You can use the pipe (````) to chain `embed` function-calls with `use` function-calls.

## 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, Microsoft Channel 9 (who host the UseR! 2016 videos), and Box.
Usage

```r
embed_user2016(
  id,
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  frameborder = 0,
  allowfullscreen = TRUE
)

embed_user2017(
  id,
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  frameborder = 0,
  allowfullscreen = TRUE
)

embed_box(
  id,
  custom_domain = getOption("vembedr.box_custom_domain"),
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  frameborder = 0,
  allowfullscreen = TRUE
)

embed_channel9(
  id,
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  frameborder = 0,
  allowfullscreen = TRUE
)

embed_msstream(
  id,
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  query = NULL
)

embed_vimeo(
  id,
```
embed

width = NULL,
height = 300,
ratio = c("16by9", "4by3"),
frameborder = 0,
allowfullscreen = TRUE,
query = NULL,
fragment = NULL
)

embed_youtube(
  id,
  width = NULL,
  height = 300,
  ratio = c("16by9", "4by3"),
  frameborder = 0,
  allowfullscreen = TRUE,
  query = NULL
)

Arguments

id character, identifier provided by the service
width numeric, width of iframe (px)
height numeric, height of iframe (px)
ratio character, indicates aspect ratio for the <iframe/>
frameborder numeric, size of frame border (px)
allowfullscreen

Arguments

custom_domain character, (used by Box) name of Box-instance to use. It can be useful to use
getOption("vembedr.box_custom_domain") if you are using a corporate instance of Box. If NULL, it will use the standard Box instance.
query list, 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
Box https://developer.box.com/docs/box-embed#section-build-box-embed-programatically
Microsoft Stream https://docs.microsoft.com/en-us/stream/portal-embed-video
**embed_url**

**Value**

Object with S3 class vembedr_embed

**See Also**

use_start_time()

**Examples**

```r
eembed_url("https://youtu.be/1-vcErOPoFQ?t=28s")
```

---

**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 works for all the services supported by the `embed()` family of functions.

**Usage**

```r
embed_url(url)
```

**Arguments**

- `url` character, URL of web-page for video

**Details**

This function calls `suggest_embed()` then parses and evaluates the code.

**Value**

Object with S3 class vembedr_embed

**See Also**

suggest_embed()

**Examples**

```r
eembed_url("https://youtu.be/1-vcErOPoFQ?t=28s")
```
get_service  

Determine service based on URL

**Description**

Determine service based on URL

**Usage**

```r
get_service(url)
```

**Arguments**

`url` character, URL of web-page for video

**Value**

character identifying the video service

**Examples**

```r
def get_service(url):
    # do something with url
```

```
get_service("https://youtu.be/1-vcEr0PofQ?t=28s")
```

---

**rickroll**  

Embed 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**

```r
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
use_align

Examples
rickroll_vimeo()
rickroll_youtube()
rickroll_channel9()

use_align          Align horizontally

Description
Use this function to specify the horizontal alignment of the iframe within the enclosing div.

Usage
use_align(embed, align = c("left", "right", "center", "justified"))

Arguments
embed vembedr_embed object, created using an embed() function
align character, indicates type of alignment

Value
Object with S3 class vembedr_embed

use_bs-responsive  Make size responsive

Description
If your HTML page includes Twitter Bootstrap 3, you can use this function to make the size of the iframe responsive within the enclosing div.

Usage
use_bs responsive(embed)

Arguments
embed vembedr_embed object, created using an embed() function

Value
Object with S3 class vembedr_embed
**use_rounded**

*Make rounded corners*

**Description**

You can use this function to make rounded corners for the enclosing `<div>`.

**Usage**

```r
use_rounded(embed, radius = NULL)
```

**Arguments**

- `embed` - `vembedr_embed` object, created using an `embed()` function
- `radius` - numeric or character, css property for the border-radius for the `<iframe/>`. Numeric values will be interpreted as number of pixels.

**Value**

Object with S3 class `vembedr_embed`

---

**use_start_time**

*Specify start time*

**Description**

This function provides you a consistent way to specify the start time, regardless of the service. Please note that Box does not provide a means to specify the start time.

**Usage**

```r
## S3 method for class 'vembedr_embed_box'
use_start_time(embed, ...) 
```

```r
## S3 method for class 'vembedr_embed_channel9'
use_start_time(embed, start_time, is_paused = TRUE, ...)
```

```r
## S3 method for class 'vembedr_embed_msstream'
use_start_time(embed, start_time, ...)
```

```r
## S3 method for class 'vembedr_embed_vimeo'
use_start_time(embed, start_time, ...)
```

```r
## S3 method for class 'vembedr_embed_youtube'
```
use_start_time(embed, start_time, ...)
use_start_time(embed, ...)

## Default S3 method:
use_start_time(embed, ...)

### Arguments
- **embed**
  vembedr_embed object, created using an `embed()` function
- **...**
  generic arguments to pass through
- **start_time**
  numeric (seconds), or character (e.g. "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.

### Value
Object with S3 class `vembedr_embed`

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

---

### Description
Knowledge of these classes is not needed for day-to-day use. Rather, it is a bookkeeping device used to make it clearer to a developer how to add a new service.
Details

We use S3 classes to distinguish an embed object, and to denote which service it uses. Objects of these classes are created by `embed_url()` and each service’s embed function.

**vembedr_embed**

- base class for all services
- HTML `<div>`
- contains the embed code

There is an additional class attached according to the service:

- `vembedr_embed_youtube`
- `vembedr_embed_youtube_short`
- `vembedr_embed_vimeo`
- `vembedr_embed_channel9`
- `vembedr_embed_box`
- `vembedr_embed_msstream`

To support parsing, there is an internal S3 class attached to the URL being processed. It is named according to the service:

- `vembedr_url_youtube`
- `vembedr_url_youtube_short`
- `vembedr_url_vimeo`
- `vembedr_url_channel9`
- `vembedr_url_box`
- `vembedr_url_msstream`
Index

* embed
  embed, 2
  embed(), 5–9
  embed_box (embed), 2
  embed_channel9 (embed), 2
  embed_msstream (embed), 2
  embed_url, 5
  embed_url(), 2, 10
  embed_user2016 (embed), 2
  embed_user2017 (embed), 2
  embed_vimeo (embed), 2
  embed_youtube (embed), 2
  embed_youtube(), 2

  get_service, 6

  rickroll, 6
  rickroll_channel9 (rickroll), 6
  rickroll_vimeo (rickroll), 6
  rickroll_youtube (rickroll), 6

  suggest_embed(), 5

  tags, 6

  use_align, 7
  use_bs_responsive, 7
  use_rounded, 8
  use_rounded(), 2
  use_start_time
    (use_start_time.vembedr_embed_box), 8
  use_start_time(), 2, 5
  use_start_time.vembedr_embed_box, 8

  vembedr-package, 2
  vembedr-s3-classes, 9