Package ‘rintrojs’

May 29, 2019

Type Package
Title Wrapper for the 'Intro.js' Library
Version 0.2.2
Description A wrapper for the 'Intro.js' library (For more info: <http://www.introjs.com>). This package makes it easy to include step-by-step introductions, and clickable hints in a 'Shiny' application. It supports both static introductions in the UI, and programmatic introductions from the server-side.
License AGPL-3
LazyData TRUE
Imports shiny, jsonlite
Depends R (>= 3.0.0)
RoxygenNote 6.1.1
URL https://github.com/carlганz/rintrojs
BugReports https://github.com/carlганz/rintrojs/issues
Suggests testthat, covr
NeedsCompilation no
Author Carl Ganz [aut, cre],
Afshin Mehrabani [ctb, cph] (intro.js in javascript/introjs)
Maintainer Carl Ganz <carlganz@gmail.com>
Repository CRAN
Date/Publication 2019-05-29 21:30:04 UTC

R topics documented:

introBox ................................................................. 2
introjs ................................................................. 3
introjsUI ............................................................... 4
readCallback ......................................................... 5

Index 6
**introBox**

Generate intro elements in UI

**Description**

Wrap `introBox` around elements you want to include in introduction. Use `data.step` to order the boxes and `data.intro` to specify the comment in the introduction.

**Usage**

```r
introBox(..., data.step, data.intro, data.hint, 
  data.position = c("bottom", "auto", "top", "left", "right", "bottom", 
    "bottom-left_aligned", "bottom-middle-aligned", "bottom-right-aligned", 
    "auto")
```

**Arguments**

- `...`: Elements in introduction element
- `data.step`: a number indicating its spot in the order in the intro
- `data.intro`: text for introduction
- `data.hint`: text for clickable hints
- `data.position`: position of intro

**See Also**

`introjsUI()` `introjs()`

**Examples**

```r
## Not run:
library(introjs)
library(shiny)
ui <- shinyUI(fluidPage(
  introjsUI(), # must include in UI
  mainPanel(
    introBox(
      tableOutput("mtcars"), 
      data.step = 1, 
      data.intro = "This is the table" ),
    introBox(
      actionButton("btn","Intro"), 
      data.step = 2, 
      data.intro = "This is the button" 
    )
  ),
))
server <- shinyServer(function(input, output, session) {
  output$mtcars <- renderTable(
```
Introduction via the intro.js library

Usage

```r
introjs(session, options = list(), events = list())
```

```r
hintjs(session, options = list(), events = list())
```

Arguments

- `session`: the Shiny session object (from the server function of the Shiny app)
- `options`: List of options to be passed to intro.js
- `events`: List of text that are the body of a Javascript function. Must wrap in `i()`

Note

For documentation on intro.js options and events, see [https://github.com/usablica/intro.js/wiki/Documentation](https://github.com/usablica/intro.js/wiki/Documentation).

See Also

`introjsUI()` `introBox()`

Examples

```r
## Not run:
library(rintrojs)
library(shiny)
u <- shinyUI(fluidPage(
  introjsUI(), # must include in UI
  mainPanel(
    introBox(
      tableOutput("mtcars"),
      data.step = 1,
```
introjsUI

Set up Shiny app to use intro.js

Description

This function must be called from a Shiny app’s UI in order to use the package.

Usage

`introjsUI(includeOnly = FALSE, cdn = FALSE, version = "2.5.0")`

Arguments

- `includeOnly` Only include intro.js files. For users who will write their own javascript
- `cdn` Indicate whether to include intro.js files from CDN
- `version` Specify intro.js version to use from cdn

Examples

```r
## Not run:
library(introjs)
library(shiny)

shinyApp(

```
ui = fluidPage(
  introjsUI(), # must include in UI
  actionButton("btn", "Click me")
),
server = function(input, output, session) {
  observeEvent(input$btn, {
    intro <- data.frame(element="#btn",
                        intro="In Codd we trust")
    introjs(session, options = list(steps=intro))
  })
})

## End(Not run)

---

**readCallback**

Read a JS callback function into rintrojs

**Description**

Reads a JS callback function into rintrojs

**Usage**

readCallback(funname = c("switchTabs"))

**Arguments**

- **funname** The name of the function you want to use. Options include:
  
  - **switchTabs** This function is intended to be passed to IntroJs’s `onbeforechange` method. It will switch the currently active tab in your Shiny app to be the one containing the next element in your tour (this function is called by IntroJs right before it moves to the next element). Try running `shiny::runApp(system.file("examples", "switchTabs.R"))` to see an example.

**Value**

A string containing the body of a callback function

**Examples**

```r
## Not run:
introjs(session, events = list(onbeforechange = readCallback("switchTabs")))

## End(Not run)
```
Index

hintjs (introjs), 3
I(), 3
introBox, 2
introBox(), 3
introjs, 3
introjs(), 2
introjsUI, 4
introjsUI(), 2, 3
readCallback, 5