

Package: dragulaR (via r-universe)

November 1, 2024

Type Package

Title Drag and Drop Elements in 'Shiny' using 'Dragula Javascript Library'

Version 0.3.2

Maintainer Zygmunt Zawadzki <zygmunt@zstat.pl>

Description Move elements between containers in 'Shiny' without explicitly using 'JavaScript'. It can be used to build custom inputs or to change the positions of user interface elements like plots or tables.

License GPL-2

LazyData TRUE

Depends htmlwidgets, shiny

Imports shinyjs, V8

RoxygenNote 7.1.1

Suggests covr, testthat

BugReports <https://github.com/zzawadz/dragulaR/issues>

URL <https://dragular.zstat.pl/>

Repository <https://zzawadz.r-universe.dev>

RemoteUrl <https://github.com/zzawadz/dragular>

RemoteRef HEAD

RemoteSha 2f42cce185a2b358941cbb759bd4c853b3928eec

Contents

dragula	2
dragulaOutput	2
dragulaValue	3
useDragulajs	4

Index	5
--------------	----------

dragula	<i>Register containers to dragula.</i>
---------	--

Description

Create dragula instace to allow moving around elements of the registered containers.

Usage

```
dragula(x, ...)
```

Arguments

x	vector of containers ids. Their's elements will become draggable.
...	additonal arguments passed to dragula JS as options. E.g. id will be an id to read from in shiny.

Value

Return htmlWidget. Should be used only inside shiny ui.

Examples

```
if(interactive()) {  
  path <- system.file("apps/example01-dragula", package = "dragulaR")  
  runApp(path, display.mode = "showcase")  
}
```

dragulaOutput	<i>Dragula widget.</i>
---------------	------------------------

Description

Create dragula widget.

Usage

```
dragulaOutput(outputId)
```

```
renderDragula(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	output variable to read from.
expr	An expression that generates a dragula object.
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

Value

RETURN_DESCRIPTION

Examples

```
if(interactive()) {  
  path <- system.file("apps/example02-input", package = "dragulaR")  
  runApp(path, display.mode = "showcase")  
}
```

dragulaValue	<i>Format dragula input object.</i>
--------------	-------------------------------------

Description

This is a utility function for better formatting dragula's input.

Usage

```
dragulaValue(x)
```

Arguments

x	dragula input.
---	----------------

Examples

```
## Not run:  
# Example call:  
dragulaValue(input$dragula)  
  
## End(Not run)
```

`useDragulajs`*Register dragulaR's js functions for refreshing dragula object.*

Description

Register dragulaR's js functions for refreshing dragula object.

Usage

```
useDragulajs()
```

Examples

```
## Not run:  
# See example for more details  
library(dragulaR)  
runApp(  
  system.file("apps/example05-dragula-dynamic-elements", package = "dragulaR"),  
  display.mode = "showcase")  
  
## End(Not run)
```

Index

`dragula`, [2](#)

`dragulaOutput`, [2](#)

`dragulaValue`, [3](#)

`renderDragula (dragulaOutput)`, [2](#)

`useDragulajs`, [4](#)