

Package ‘shinytabconstructor’

February 23, 2018

Type Package

Title Opinionated Shiny Structure

Version 0.4.3

Author McClelland Legge <McClelland.Kemp@iriworldwide.com> [aut, cre]

Maintainer <McClelland.Kemp@iriworldwide.com>

Description A package to structure and run shiny code in a way that makes it comprehensible when codebases grow large

License MIT-2

Encoding UTF-8

LazyData true

Suggests testthat

Imports tools,
shiny,
shinydashboard,
purrr,
utils,
brew,
rprojroot

RoxygenNote 6.0.1

R topics documented:

addTab	2
checkDuplicateTabNames	2
checkNamingConventions	3
deleteTab	3
extractTabName	4
filesInTreeStructure	4
findTab	5
getChildTabs	5
getExistingTabs	6
getServerLocation	6
getTabLocation	6
includeR	7
initTabConstructor	7
makeMenu	7

menuItemList	8
serverFunc	9
tabItemList	9
thisTabName	9

Index	10
--------------	-----------

addTab	<i>Create a new tab</i>
--------	-------------------------

Description

Create a new tab

Usage

addTab(name, parent = "ui", active = TRUE, open = TRUE)

Arguments

name	A character string. What to call the tab? Cannot start with a number.
parent	A character string. The name of the tab the new parent should go under. Defaults to 'ui' which indicates the tab should be accessible on the top level of the main menu
active	A Boolean. Should the .yaml be initialized to immediately include the tab in the app?
open	A Boolean. Should RStudio attempt to open the tab you just created?

checkDuplicateTabNames	<i>Check the files against naming conventions</i>
------------------------	---

Description

Check the files against naming conventions

Usage

checkDuplicateTabNames(tocheck = NULL)

Arguments

tocheck	An optional character vector of names to check against the existing ones
---------	--

`checkNamingConventions`*Check the naming conventions of the app files*

Description

Check the naming conventions of the app files

Usage

```
checkNamingConventions(files, criteria = "[^a-z0-9_\\.\\/]")
```

Arguments

<code>files</code>	A character vector of file names
<code>criteria</code>	A regex expression to definite the allowable characters

Value

TRUE

Examples

```
checkNamingConventions(c("01-desc.R", "a.good.name.R", "another_acceptable.name.R"))
```

`deleteTab`*Remove an existing tab*

Description

Remove an existing tab

Usage

```
deleteTab(name, cascade = FALSE)
```

Arguments

<code>name</code>	A character string. What to call the tab? Cannot start with a number.
<code>cascade</code>	A boolean, should it delete all tabs below it?

extractTabName	<i>Extract the tab name from the file name</i>
----------------	--

Description

Extract the tab name from the file name

Usage

```
extractTabName(path)
```

Arguments

path	An .R or .yaml file path
------	--------------------------

Value

A character string

Examples

```
extractTabName("a.dot.delimited.name.tabname.R")
```

filesInTreeStructure	<i>Determine a directory file structure</i>
----------------------	---

Description

Determine a directory file structure

Usage

```
filesInTreeStructure(file, pattern = NULL)
```

Arguments

file	The root directory or file
pattern	An optional pattern to filter out directories containing files

Value

A list

findTab	<i>Find an existing tab</i>
---------	-----------------------------

Description

Find an existing tab

Usage

```
findTab(name)
```

Arguments

name	The name of the tab to find the location for
------	--

Value

A filepath of the location of that tab

getChildTabs	<i>Get the names of the tabs that are nested under a given tab</i>
--------------	--

Description

Get the names of the tabs that are nested under a given tab

Usage

```
getChildTabs(name)
```

Arguments

name	A character string, the name of the tab
------	---

Value

A character vector

getExistingTabs	<i>Find the existing tab names</i>
-----------------	------------------------------------

Description

Find the existing tab names

Usage

```
getExistingTabs()
```

Value

A character vector

Examples

```
getExistingTabs()
```

getServerLocation	<i>Find the physical location of the server file within project</i>
-------------------	---

Description

Find the physical location of the server file within project

Usage

```
getServerLocation()
```

Value

A character string: The fully qualified file path to the server folder

getTabLocation	<i>Find the physical location of the tabs within the project</i>
----------------	--

Description

Find the physical location of the tabs within the project

Usage

```
getTabLocation()
```

Value

A character string: The fully qualified file path to the tabs folder

includeR	<i>Recursively source R scripts and return the values</i>
----------	---

Description

Recursively source R scripts and return the values

Usage

```
includeR(path, single_as_list = FALSE, ...)
```

Arguments

path	A path to an R script
...	Additional argument to <code>list.files</code>
single_as_list	A Boolean. Should a single sourced value be returned as list? Useful for server functions

Value

A list if more than one file sourced, otherwise the value itself

initTabConstructor	<i>Initialize the framework</i>
--------------------	---------------------------------

Description

Initialize the framework

Usage

```
initTabConstructor()
```

makeMenu	<i>Make a menu from a configured hierarchical list</i>
----------	--

Description

Make a menu from a configured hierarchical list

Usage

```
makeMenu(.list)
```

Arguments

.list	A (possibly) nested list of the .yaml files defining a menu structure
-------	---

Value

A list of html menu objects

menuItemList

*Create a menu item***Description**

Create a menu item

Usage

```
menuItemList(text, .list, icon = NULL, badgeLabel = NULL,
  badgeColor = "green", tabName = NULL, href = NULL, newtab = TRUE,
  selected = NULL, expandedName = as.character(gsub("[[:space:]]", "",
text)), startExpanded = FALSE)
```

Arguments

text	Text to show for the menu item.
.list	An optional list of menuItemSubItems
icon	An icon tag, created by icon . If NULL, don't display an icon.
badgeLabel	A label for an optional badge. Usually a number or a short word like "new".
badgeColor	A color for the badge. Valid colors are listed in validColors .
tabName	The name of a tab that this menu item will activate. Not compatible with href.
href	An link address. Not compatible with tabName.
newtab	If href is supplied, should the link open in a new browser tab?
selected	If TRUE, this menuItem or menuItemSubItem will start selected. If no item have selected=TRUE, then the first menuItem will start selected.
expandedName	A unique name given to each menuItem that serves to indicate which one (if any) is currently expanded. (This is only applicable to menuItems that have children and it is mostly only useful for bookmarking state.)
startExpanded	Should this menuItem be expanded on app startup? (This is only applicable to menuItems that have children, and only one of these can be expanded at any given time).
id	For sidebarMenu, if id is present, this id will be used for a Shiny input value, and it will report which tab is selected. For example, if id="tabs", then input\$tabs will be the tabName of the currently-selected tab. If you want to be able to bookmark and restore the selected tab, an id is required.

Details

Exactly the same as [menuItem](#) except that the [menuItemSubItems](#) can be passed in a list. All creational credit to the maintainers of that package

serverFunc	<i>Source the server functions</i>
------------	------------------------------------

Description

Source the server functions

Usage

```
serverFunc(input, output, session)
```

tabItemList	<i>Form a tab structure from one or more tab items</i>
-------------	--

Description

Form a tab structure from one or more tab items

Usage

```
tabItemList(items)
```

Arguments

items	One or more tabItem in a list
-------	---

Value

A html div

thisTabName	<i>Get name of tab based on file name</i>
-------------	---

Description

Get name of tab based on file name

Usage

```
thisTabName()
```

Value

A character string or NULL

Index

addTab, [2](#)

checkDuplicateTabNames, [2](#)
checkNamingConventions, [3](#)

deleteTab, [3](#)

extractTabName, [4](#)

filesInTreeStructure, [4](#)
findTab, [5](#)

getChildTabs, [5](#)
getExistingTabs, [6](#)
getServerLocation, [6](#)
getTabLocation, [6](#)

icon, [8](#)
includeR, [7](#)
initTabConstructor, [7](#)

makeMenu, [7](#)
menuItem, [8](#)
menuItemList, [8](#)
menuSubItem, [8](#)

serverFunc, [9](#)

tabItem, [9](#)
tabItemList, [9](#)
thisTabName, [9](#)

validColors, [8](#)