Package 'miniUI'

May 18, 2018

Type Package

Title Shiny UI Widgets for Small Screens

Version 0.1.1.1

Description Provides UI widget and layout functions for writing Shiny apps that work well on small screens.

License GPL-3

LazyData TRUE

Imports shiny (>= 0.13), htmltools (>= 0.3), utils

RoxygenNote 5.0.1

NeedsCompilation no

Author Joe Cheng [cre, aut],
RStudio [cph]

Maintainer Joe Cheng <joe@rstudio.com>

Repository CRAN

Date/Publication 2018-05-18 18:37:18 UTC

R topics documented:

miniButtonBlock *Create a block-level button container*

Description

Creates a full-width container for one or more buttons. The horizontal space will be evenly divided among any buttons that are added.

Usage

```
miniButtonBlock(..., border = "top")
```

Arguments

...	One or more actionButton or downloadButton objects.
border	Zero or more of c("top", "bottom"), indicating which sides should have borders, if any.

Details

When using miniButtonBlock with a miniTabstripPanel, consider passing the miniButtonBlock to miniTabstripPanel as the between argument.

See Also

For more information, see the Designing Gadget UI article on shiny.rstudio.com.

Examples

```
library(shiny)

miniButtonBlock(
  actionButton("reset", "Reset to defaults"),
  actionButton("clear", "Clear all")
)
```

miniContentPanel *Create a content panel*

Description

Creates a panel for containing arbitrary content within a flex box container. This is mainly useful within miniPage or a miniTabPanel. You can use miniContentPanel to introduce padding and/or scrolling, but even if padding/scrolling aren't needed, it's a good idea to wrap your custom content into miniContentPanel as it fixes some odd behavior with percentage-based heights.

Usage

```
miniContentPanel(..., padding = 15, scrollable = TRUE)
```

Arguments

...	UI objects to be contained in the miniContentPanel. A single htmlwidget or plotOutput with height="100%" works well, as do fillRow/fillCol.
padding	Amount of padding to apply. Can be numeric (in pixels) or character (e.g. "3em").
scrollable	If TRUE, then content large enough to overflow the miniContentPanel will make scrollbars appear.

See Also

For more information, see the Designing Gadget UI article on shiny.rstudio.com.

Examples

```
library(shiny)

miniContentPanel(padding = 0,
  plotOutput("plot", height = "100%")
)
```

---

miniPage *Page function for Shiny Gadgets*

Description

Designed to serve as the outermost function call for your gadget UI. Similar to fillPage, but always includes the Bootstrap CSS library, and is designed to contain miniTitleBar, miniTabstripPanel, miniContentPanel, etc.

Usage

```
miniPage(..., title = NULL, theme = NULL)
```

Arguments

...	Elements to include within the page.
title	The title to use for the browser window/tab (it will not be shown in the document).
theme	URL to alternative Bootstrap stylesheet.

See Also

For more information, see the Designing Gadget UI article on shiny.rstudio.com.

`miniTabstripPanel` *Create a tabstrip panel*

Description

`miniTabstripPanel` is a tabstrip panel that contains `miniTabPanel` elements. Similar to `tabsetPanel`, but optimized for small page sizes like mobile devices or the RStudio Viewer pane.

Usage

```
miniTabstripPanel(..., id = NULL, selected = NULL, between = NULL)

miniTabPanel(title, ..., value = title, icon = NULL)
```

Arguments

`...`	For `miniTabstripPanel`, `miniTabPanel` elements to include in the tabset. For `miniTabPanel`, UI elements to include within the tab.
`id`	If provided, you can use `input$id` in your server logic to determine which of the current tabs is active. The value will correspond to the `value` argument that is passed to `miniTabPanel`.
`selected`	The `value` (or, if none was supplied, the `title`) of the tab that should be selected by default. If `NULL`, the first tab will be selected.
`between`	A tag or list of tags that should be inserted between the content (above) and tabstrip (below).
`title`	Display title for tab.
`value`	The value that should be sent when `miniTabstripPanel` reports that this tab is selected. If omitted and `miniTabstripPanel` has an `id`, then the tab's title will be used as the value.
`icon`	Icon to appear on the tab; see `icon`.

See Also

For more information, see the Designing Gadget UI article on shiny.rstudio.com.

Examples

```
library(shiny)

miniTabstripPanel(
  miniTabPanel("Data", icon = icon("table"),
    selectInput("dataset", "Data set", ls("package:datasets"))),
  miniTabPanel("Subset", icon = icon("sliders"),
    uiOutput("subset_ui")
  )
)
```

`miniTitleBar` *Create a title bar*

Description

Creates a title bar for a Shiny app or Shiny Gadget. Intended to be used with `miniPage`. Title bars contain a title, and optionally, a `miniTitleBarButton` on the left and/or right sides.

Usage

```
miniTitleBar(title, left = NULL, right = NULL)

gadgetTitleBar(title, left = miniTitleBarCancelButton(),
  right = miniTitleBarButton("done", "Done", primary = TRUE))

miniTitleBarButton(inputId, label, primary = FALSE)

miniTitleBarCancelButton(inputId = "cancel", label = "Cancel",
  primary = FALSE)
```

Arguments

`title`	The title of the gadget. If this needs to be dynamic, pass `textOutput` with `inline = TRUE`.
`left`	The `miniTitleBarButton` to put on the left, or `NULL` for none.
`right`	The `miniTitleBarButton` to put on the right, or `NULL` for none.
`inputId`	The `input` slot that will be used to access the button.
`label`	The text label to display on the button.
`primary`	If `TRUE`, render the button in a bold color to indicate that it is the primary action of the gadget.

Details

`gadgetTitleBar` is a `miniTitleBar` with different defaults: a Cancel button on the left and a Done button on the right. By default, `runGadget` will handle the Cancel button by closing the gadget and raising an error, but the `Done` button must be handled by the gadget author using `observeEvent(input$done, {...})`.

`miniTitleBarCancelButton` is like `miniTitleBarButton`, but the user can also invoke it by hitting the Escape key.

See Also

For more information, see the Designing Gadget UI article on shiny.rstudio.com.

Examples

```
miniTitleBar("My App",
  left = miniTitleBarButton("prev", "Previous"),
  right = miniTitleBarButton("next", "Next")
)
```

Index