Package: react (via r-universe)

June 8, 2024

Title Reactivity Helper for 'shiny'

Version 2024.1.0.9000

Description Tools to help with 'shiny' reactivity. The 'react' function offers alternative ways to call reactive expressions to better identify them in the server code.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports cli, glue, rlang

URL https://github.com/tadascience/react, https://react.tada.science/

BugReports https://github.com/tadascience/react/issues

Repository https://tadascience.r-universe.dev

RemoteUrl https://github.com/tadascience/react

RemoteRef HEAD

RemoteSha 74b6dea2517e47efa7afcaa4facf4420b9dbbc15

Contents

react *Reactivity helper*

Description

The react object gives alternative syntax to call shiny reactive expressions.

Usage

```
react(x)
```

Arguments

x   the reactive call

Details

The benefit is that it makes them easier to spot in your code.

Examples

```
# This works by invoking the function from the parent environment
# with no arguments ...
foo <- function() {
  42
}
react$foo
react[foo]
react[foo()]

# You can also use `react()` as a function to wrap the
# reactive call
react(foo())

# The benefit is that `react()` can also wrap `input$` calls
# so that you easily recognize reactivity

## Not run:
  # ... but it only becomes relevant when used in shiny
  # server code, e.g. this app from the shiny page
  # with react$dataInput instead of dataInput()
  server <- function(input, output) {

    dataInput <- reactive({
      getSymbols(input$symb, src = "yahoo",
          from = input$dates[1],
            to = input$dates[2],
            auto.assign = FALSE)
    })
```

```
       output$plot <- renderPlot({
          chartSeries(react$dataInput, theme = chartTheme("white"),
                type = "line", log.scale = input$log, TA = NULL)
       })
      }

## End(Not run)
```

Index