Package ‘daterangeinput’

May 10, 2023

Type Package
Title Create a Shiny Date-Range Input
Version 0.2.0
Maintainer Sebastian Gatscha <sebastian_gatscha@gmx.at>
Description A Shiny Input for date-ranges, which pops up two calendars for selecting dates, times, or predefined ranges like "Last 30 Days". It wraps the JavaScript library 'daterangepicker' which is available at <https://www.daterangepicker.com>.
License MIT + file LICENSE
Depends R (>= 3.2.0)
Imports htmltools, jsonify, utils, shiny
URL https://github.com/trafficonese/daterangepicker/,
    https://www.daterangepicker.com,
    https://trafficonese.github.io/daterangepicker/
BugReports https://github.com/trafficonese/daterangepicker/issues
Encoding UTF-8
RoxygenNote 7.2.0
Suggests testthat (>= 2.1.0), covr
NeedsCompilation no
Author Sebastian Gatscha [aut, cre]
Repository CRAN
Date/Publication 2023-05-10 12:00:06 UTC

R topics documented:

checkMaxSpan .................................................. 2
checkRanges .................................................. 2
daterangepicker ................................................. 3
daterangepickerOptions ........................................ 5
filterEMPTY .................................................. 7
checkMaxSpan

**Description**

checkMaxSpan Check the maxSpan element, for Time objects

**Usage**

checkMaxSpan(maxSpan)

**Arguments**

maxSpan The list of ranges

checkRanges

**Description**

checkRanges Check the ranges element, for Date or POSIX objects

**Usage**

checkRanges(ranges)

**Arguments**

ranges The list of ranges
**Description**

The Date Range Picker pops up two calendars for selecting dates, times, or predefined ranges like "Yesterday", "Last 30 Days", etc.

**Usage**

daterangepicker(
    inputId = NULL,
    label = "Select a Date",
    start = NULL,
    end = NULL,
    min = NULL,
    max = NULL,
    ranges = NULL,
    rangeNames = names(ranges),
    language = "en",
    style = "width:100%;border-radius:4px;text-align:center;",
    class = NULL,
    icon = NULL,
    options = daterangepickerOptions()
)

**Arguments**

- **inputId**
  The input ID

- **label**
  The label for the control, or NULL for no label.

- **start**
  The beginning date of the initially selected. Must be a Date / POSIXt or string. If NULL will default to the current day.

- **end**
  The end date of the initially selected date range. Must be a Date / POSIXt or string. If NULL will default to the current day.

- **min**
  The earliest date a user may select. Must be a Date or string

- **max**
  The latest date a user may select. Must be a Date or string

- **ranges**
  Set predefined date ranges the user can select from. Each key is the label for the range, and its value an array with two dates representing the bounds of the range. Alternatively, the labels can be specified via 'rangeNames'. If that argument is used, 'ranges' should not be named and 'rangeNames' will take precedence.

- **rangeNames**
  Optional character vector specifying the labels for predefined date ranges. If specified, it will override the names of 'ranges'.

- **language**
  The language used for month and day names. Default is "en". See the Multiple Locale Support for a list of other valid values.
style  Add CSS-styles to the input.
class  Custom class
icon   Icon to display next to the label.
options List of further options. See `daterangepickerOptions`

See Also

[www.daterangepicker.com](http://www.daterangepicker.com)

Other daterangepicker Functions: `daterangepickerOptions()`, `updateDaterangepicker()`

Examples

```r
if (interactive()) {
  library(shiny)
  library(daterangepicker)
  ## UI ##########################
  ui <- fluidPage(
    tags$head(tags$style(".myclass {background-color: #96dafb;}")),
    daterangepicker(
      inputId = "daterange",
      label = "Pick a Date",
      start = Sys.Date() - 30, end = Sys.Date(),
      max = Sys.Date(),
      language = "en",
      ranges = list("Today" = Sys.Date(),
                    "Yesterday" = Sys.Date() - 1,
                    "Last 3 days" = c(Sys.Date() - 2, Sys.Date()),
                    "Last 7 days" = c(Sys.Date() - 6, Sys.Date())),
      style = "width:100%; border-radius:4px",
      class = "myclass",
      icon = icon("calendar")
    ),
    verbatimTextOutput("print"),
    actionButton("act", "Update Daterangepicker"),
  )
  ## SERVER ##########################
  server <- function(input, output, session) {
    output$print <- renderPrint({
      req(input$daterange)
      input$daterange
    })
    observeEvent(input$act, {
      updateDaterangepicker(
        session, "daterange",
        start = Sys.Date(),
        end = Sys.Date() - 100,
        ranges = list(
          Sys.Date(), Sys.Date() - 1,
          c(Sys.Date() - 3, Sys.Date())),
    })
  }
```
daterangepickerOptions

**Description**

Update the daterangepicker

**Usage**

daterangepickerOptions(
  minYear = NULL,
  maxYear = NULL,
  showDropdowns = TRUE,
  showCustomRangeLabel = TRUE,
  opens = c("right", "left", "center"),
  drops = c("down", "up"),
  timePicker = FALSE,
  timePickerIncrement = 1,
  timePicker24Hour = FALSE,
  timePickerSeconds = FALSE,
  showWeekNumbers = FALSE,
  showISOWeekNumbers = FALSE,
  parentEl = NULL,
  maxSpan = NULL,
  alwaysShowCalendars = FALSE,
  buttonClasses = NULL,
  applyButtonClasses = NULL,
  cancelButtonClasses = NULL,
  cancelIsClear = FALSE,
  locale = NULL,
  singleDatePicker = FALSE,
  autoApply = FALSE,
  linkedCalendars = TRUE,
autoUpdateInput = TRUE,
...
)

Arguments

minYear  The minimum year as numeric shown in the dropdowns when showDropdowns is set to TRUE
maxYear  The maximum year as numeric shown in the dropdowns when showDropdowns is set to TRUE
showDropdowns  Show year and month select boxes above calendars to jump to a specific month and year.
showCustomRangeLabel  Displays "Custom Range" at the end of the list of predefined ranges, when the ranges option is used. This option will be highlighted whenever the current date range selection does not match one of the predefined ranges. Clicking it will display the calendars to select a new range.
opens  Whether the picker appears aligned to the left, to the right, or to the center under the HTML element it’s attached to.
drops  Whether the picker appears below (down) or above (up) the HTML element it’s attached to. Default is down
timePicker  Adds select boxes to choose times in addition to dates.
timePickerIncrement  Increment of the minutes selection list for times (i.e. 30 to allow only selection of times ending in 0 or 30).
timePicker24Hour  Use 24-hour instead of 12-hour times, removing the AM/PM selection.
timePickerSeconds  Show seconds in the timePicker.
showWeekNumbers  Show localized week numbers at the start of each week on the calendars.
showISOWeekNumbers  Show ISO week numbers at the start of each week on the calendars.
parentEl  jQuery selector of the parent element that the date range picker will be added to, if not provided this will be ‘body’
maxSpan  The maximum span between the selected start and end dates as List or table. You can provide any object the moment library would let you add to a date.
alwaysShowCalendars  Normally, if you use the ranges option to specify pre-defined date ranges, calendars for choosing a custom date range are not shown until the user clicks "Custom Range". When this option is set to TRUE, the calendars for choosing a custom date range are always shown instead.
buttonClasses  CSS class names that will be added to both the apply and cancel buttons.
applyButtonClasses  CSS class names that will be added only to the apply button.
cancelButtonClasses  
CSS class names that will be added only to the cancel button.

cancelIsClear  
If TRUE, will treat the Cancel Button like a Clear Button.

locale  
Allows you to provide localized strings for buttons and labels, customize the date format, and change the first day of week for the calendars. See the examples in `./inst/examples/`

singleDatePicker  
Show only a single calendar to choose one date, instead of a range picker with two calendars. The start and end dates provided to your callback will be the same single date chosen.

autoApply  
Hide the apply and cancel buttons, and automatically apply a new date range as soon as two dates are clicked.

linkedCalendars  
When enabled, the two calendars displayed will always be for two sequential months (i.e. January and February), and both will be advanced when clicking the left or right arrows above the calendars. When disabled, the two calendars can be individually advanced and display any month/year.

autoUpdateInput  
Indicates whether the date range picker should automatically update the value of the `<input>` element it’s attached to at initialization and when the selected dates change.

...  
Further arguments passed to 'daterangepicker', like `isInvalidDate` or `isCustomDate`.

### See Also

[www.daterangepicker.com](http://www.daterangepicker.com)

Other daterangepicker Functions: `daterangepicker()`, `updateDaterangepicker()`

---

### filterEMPTY

**filterEMPTY** Filter empty elements of a list

<table>
<thead>
<tr>
<th>Description</th>
<th>filterEMPTY Filter empty elements of a list</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage</td>
<td>filterEMPTY(x)</td>
</tr>
<tr>
<td>Arguments</td>
<td>x The list of values</td>
</tr>
</tbody>
</table>

---
makeInput

makeInput Make the input div-tag

Description
makeInput Make the input div-tag

Usage
makeInput(label, inputId, class, icon, style, options)

Arguments
- label: The label of the daterangepicker
- inputId: The inputId of the daterangepicker
- class: The class of the daterangepicker
- icon: The icon of the daterangepicker
- style: The style of the daterangepicker
- options: The options of the daterangepicker

makeLabel

makeLabel Make the label

Description
makeLabel Make the label

Usage
makeLabel(label, inputId)

Arguments
- label: The label of the daterangepicker
- inputId: The inputId of the daterangepicker
Description

Change the start and end values of a daterangepicker on the client

Usage

updateDaterangepicker(
  session,
  inputId,
  label = NULL,
  start = NULL,
  end = NULL,
  min = NULL,
  max = NULL,
  icon = NULL,
  options = NULL,
  ranges = NULL,
  rangeNames = NULL,
  style = NULL,
  class = NULL
)

Arguments

- **session**: The session object passed to function given to shinyServer.
- **inputId**: The input ID
- **label**: The label for the control, or NULL for no label.
- **start**: The beginning date of the initially selected. Must be a Date / POSIXt or string. If NULL will default to the current day.
- **end**: The end date of the initially selected date range. Must be a Date / POSIXt or string. If NULL will default to the current day.
- **min**: The earliest date a user may select. Must be a Date or string
- **max**: The latest date a user may select. Must be a Date or string
- **icon**: Icon to display next to the label.
- **options**: List of further options. See daterangepickerOptions
- **ranges**: Set predefined date ranges the user can select from. Each key is the label for the range, and its value an array with two dates representing the bounds of the range. Alternatively, the labels can be specified via 'rangeNames'. If that argument is used, 'ranges' should not be named and 'rangeNames' will take precedence.
- **rangeNames**: Optional character vector specifying the labels for predefined date ranges. If specified, it will override the names of 'ranges'.
- **style**: Add CSS-styles to the input.
- **class**: Custom class
See Also

Other daterangepicker Functions: `daterangepickerOptions()`, `daterangepicker()`
Index

* daterangepicker Functions
  daterangepicker, 3
  daterangepickerOptions, 5
  updateDaterangepicker, 9

checkMaxSpan, 2
checkRanges, 2

daterangepicker, 3, 7, 10
daterangepickerOptions, 4, 5, 9, 10

filterEMPTY, 7

makeInput, 8
makeLabel, 8

updateDaterangepicker, 4, 7, 9