# Package 'shinycroneditor'

September 23, 2024

**Title** 'shiny' Cron Expression Input Widget

**Version** 1.0.0

**Description** A widget for 'shiny' apps to handle schedule expression input, using the 'cron-expression-input' JavaScript component. Note that this does not edit the 'crontab' file, it is just an input element for the schedules. See <[https://github.com/DatalabFabriek/shinycroneditor/blob/main/inst/examples/shiny-app.R](https://github.com/DatalabFabriek/shinycroneditor/blob/main/inst/examples/shiny-app.R)> for an example implementation.

**Imports** htmltools, htmlwidgets

**Suggests** shiny

**License** GPL (>= 3)

**Encoding** UTF-8

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** Harmen van der Veer [aut, cre]

**Maintainer** Harmen van der Veer <harmen@datalab.nl>

**Repository** CRAN

**Date/Publication** 2024-09-23 13:00:06 UTC

# Contents

---

cron                 *Cron editor, creates a 'htmlwidgets' object, for use in a 'shiny' dashboard*

---

## Description

Cron editor, creates a 'htmlwidgets' object, for use in a 'shiny' dashboard

1

**Usage**

```
cron(schedule, width = NULL, height = NULL, elementId = NULL)
```

**Arguments**

| | |
|---|---|
| schedule | Any valid cron schedule, for example "0 6 * * *" |
| width | How wide should the editor be? Defaults to NULL, meaning 100% |
| height | How high should the editor be? Defaults to NULL, meaning 32px |
| elementId | Optionally specifiy the ID of the element |

**Value**

Returns a 'htmlwidgets' object that can be used in a 'shiny' app

**Examples**

```
# A simple cron editor in a Shiny app
if (interactive()) {
library(shiny)
library(shinycroneditor)

ui <- fluidPage(
titlePanel("Cron Expression Input Widget"),
mainPanel(
  shinycroneditor::cronOutput("cronschedule1",
                              label = "Choose your first schedule",
                              language = "en-US"),

    shiny::div(
      "Your first chosen schedule is: ",
      verbatimTextOutput("cronExpression1")
    ),

    shinycroneditor::cronOutput("cronschedule2",
                                label = "Choose your second schedule",
                                language = "en-US"),

    shiny::div(
      "Your chosen second schedule is: ",
      verbatimTextOutput("cronExpression2")
    )
 )
)

server <- function(input, output, session) {

 output$cronschedule1 <- shinycroneditor::renderCron({
   shinycroneditor::cron("0 6 * * *")
 })

 output$cronExpression1 <- renderPrint({
```

```
    input$cronschedule1
  })

  output$cronschedule2 <- shinycroneditor::renderCron({
    shinycroneditor::cron("30 1,3,7 * * *")
  })

  output$cronExpression2 <- renderPrint({
    input$cronschedule2
  })

}

shinyApp(ui, server)
}
```

---

croneditor-shiny            *Shiny bindings for cron editor*

---

### Description

Output and render functions for using cron editor within 'shiny' applications and interactive 'rmarkdown' documents.

### Usage

```
cronOutput(
  outputId,
  label = NULL,
  language = "en-US",
  width = "100%",
  height = NULL
)

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

### Arguments

| | |
|---|---|
| outputId | output variable to read from |
| label | The label to show, just like in a regular Shiny input element. Set to NULL if you don't want a label. |
| language | The language the cron editor's UI will be in. Choose one of en-US, nl-NL, es-ES, zh-CN. Note that if you have multiple cron editors on one page, the last editor's language will be the one for all editors. |
| width, height | Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended. |

| expr | An expression that generates a test |
|---|---|
| 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**

An 'htmlwidgets' object for use in a shiny app, containing a placeholder for where the cron input element is rendered.

A 'htmlwidgets' object for use in a 'shiny' app as an input element

# Index