

# Package ‘shinyMatrix’

February 5, 2019

**Title** Shiny Matrix Input Field

**Version** 0.1.0

**Author** Andreas Neudecker

**Maintainer** Andreas Neudecker <andreas.neudecker@inwt-statistics.de>

**Description** Implements a custom matrix input field.

**Depends** R (>= 3.5)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** shiny, jsonlite

**Suggests** testthat, covr

**RoxygenNote** 6.1.0

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-02-05 22:00:03 UTC

## R topics documented:

matrixInput . . . . .	2
startApplication . . . . .	3
updateMatrixInput . . . . .	3
<b>Index</b>	<b>5</b>

---

matrixInput

*Create a matrix input field*


---

### Description

Use this function to create a matrix input field. Typically this would be in the ‘ui.R’ file of your shiny application. You can access the input via the ‘input\$inputId’ in the server function of the shiny application. The value will always be a matrix and contain values of class ‘class’.

### Usage

```
matrixInput(inputId, value = matrix("", 1, 1), inputClass = "",
  rows = list(), cols = list(), class = "character", paste = FALSE,
  copy = FALSE)
```

### Arguments

inputId	The input slot that will be used to access the value
value	Initial value. Should be a matrix
inputClass	class of the matrix input html element
rows	list of options to configure rows
cols	list of options to configure cols
class	Matrix will be coerced to a matrix of this class. ‘character’ and ‘numeric’ are supported
paste	enable paste functionality
copy	enable copy functionality

### Details

The parameters ‘rows’ and ‘cols’ take a list of arguments. Currently, the following arguments are supported:

**n** number of rows (is calculated from value as default)

**names** should row/colnames be displayed? The names are taken from value

**editableNames** should row/colnames be editable

**extend** Should the matrix be extended if data is entered in the last row/column?

**delta** how many blank rows/cols should be added?

**createHeader, updateHeader** name of javascript function to override default function to create/update table header. The function needs to have the table element and the data object as argument

**getHeader** same as createHeader but with table element as only argument

**Examples**

```
matrixInput(  
  "myMatrix",  
  value = diag(3),  
  rows = list(names = FALSE),  
  cols = list(names = FALSE),  
  copy = TRUE,  
  paste = TRUE  
)
```

---

startApplication	<i>Start Application</i>
------------------	--------------------------

---

**Description**

This function starts an example app from the folder 'inst'.

**Usage**

```
startApplication(app = "app", port = 4242)
```

**Arguments**

app	name of the folder in 'inst'
port	port of web application

**Examples**

```
## Not run:  
startApplication("appCopy")  
  
## End(Not run)
```

---

updateMatrixInput	<i>Update matrix input</i>
-------------------	----------------------------

---

**Description**

This function updates the matrix input from R created with 'matrixInput'. It works like the other updateXXXInput functions in shiny.

**Usage**

```
updateMatrixInput(session, inputId, value)
```

**Arguments**

session	shiny session
inputId	id of matrix input
value	new value for matrix

**Examples**

```
## Not run:  
updateMatrixInput(session, "myMatrix", diag(4))  
  
## End(Not run)
```

# Index

`matrixInput`, 2

`startApplication`, 3

`updateMatrixInput`, 3