

Package: emberui (via r-universe)

October 24, 2026

Title Interactive Inputs for Ember Notebooks

Version 0.0.0.9000

Description Sliders, number boxes, checkboxes, text boxes, dropdowns and radio buttons for Ember notebooks, alone or several together as one input. In Ember, an input assigned to a name becomes a widget, and moving it reruns the cells that read the name. Anywhere else (Rscript, the R console, knitr) each function checks its arguments and returns its default value.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository <https://jowch.r-universe.dev>

Date/Publication 2026-10-24 15:52:35 UTC

RemoteUrl <https://github.com/jowch/Ember>

RemoteRef HEAD

RemoteSha 5dfa9f98b55659ca05c0152b66b5a5c9b76063a8

RemoteSubdir inst/emberui

Contents

inputs	2
Index	4

inputs

*Interactive inputs***Description**

Each function returns a plain value: a number, TRUE/FALSE, a string, an element of choices, or for `input_list()` a named list of its inputs' values. In an Ember notebook, an input assigned to a name at the top of a cell (`n <- input_slider(1, 10)`) is shown as a widget, and moving it sets `n` and reruns the cells that read `n`. Anywhere else (Rscript, the R console, knitr), and for any other call, the function checks its arguments and returns its default.

Usage

```
input_slider(min, max, value = min, step = NULL, label = NULL)
input_number(value = 0, min = NA, max = NA, step = NA, label = NULL)
input_checkbox(value = FALSE, label = NULL)
input_text(value = "", label = NULL)
input_select(choices, selected = choices[[1]], multiple = FALSE, label = NULL)
input_radio(choices, selected = choices[[1]], label = NULL)
input_list(...)
```

Arguments

<code>min, max</code>	The range of the input. For <code>input_number()</code> , NA means no limit.
<code>value</code>	The default value. A number keeps its type: <code>input_slider(1L, 10L)</code> returns an integer.
<code>step</code>	The step between the widget's values, or NULL or NA for none. A slider's values are its steps, counted from min: a value between them, the default included, is an error. A number box takes its step as a hint and allows a value between steps.
<code>label</code>	The text shown with the widget. By default, the name the input is assigned to.
<code>choices</code>	A vector or list. The value is an element itself, not its position. Names, if present, are what the widget shows. Numbers match by value, as with <code>%in%</code> : <code>selected = 5</code> picks the choice 5L.
<code>selected</code>	The default choice: an element of choices, or with <code>multiple = TRUE</code> , a vector (or list) of them.
<code>multiple</code>	Whether several choices can be picked.
<code>...</code>	For <code>input_list()</code> : named input calls, such as <code>n = input_slider(1, 10)</code> . Each must be an input call itself.

Details

Inside Ember, the input is bound to a name only when the call is the whole right-hand side of a top-level assignment and nothing else in the cell reads or redefines that name. Only the input on the cell's last line is shown as a widget, as the cell's output; any other input shows as a line below the cell with its value, saying how to fix it. Moving an input doesn't rerun its own cell.

To move several inputs from one cell, combine them with `input_list()`: `p <- input_list(n = input_slider(1, 10), method = input_select(c("a", "b")))` shows both widgets together, and other cells read `p$n` and `p$method`. Moving either sets `p`.

Input values aren't saved with the notebook: it opens with every input at its default, the value Rscript uses. To keep a value, change the default in the code.

Value

The input's value. Outside Ember, the default.

Examples

```
n <- input_slider(10, 500, value = 100, step = 10)
method <- input_select(c(Pearson = "pearson", Spearman = "spearman"))
show <- input_checkbox(TRUE, label = "Show points")
title <- input_text("Results")
p <- input_list(n = input_slider(1, 10), method = input_select(c("a", "b")))
n
method
p$n
```

Index

`input_checkbox(inputs)`, [2](#)
`input_list(inputs)`, [2](#)
`input_number(inputs)`, [2](#)
`input_radio(inputs)`, [2](#)
`input_select(inputs)`, [2](#)
`input_slider(inputs)`, [2](#)
`input_text(inputs)`, [2](#)
`inputs`, [2](#)