

# Package ‘flashCard’

May 8, 2026

**Title** Create a Flash Card

**Version** 0.1.0

**Description** Create a flip over style Flash Card with desired data frame for Shiny application.

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**License** MIT + file LICENSE

**Imports** htmlwidgets, jsonlite

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Author** Jiena McLellan [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-5578-088X>>),  
Michael Condouris [ctb] (ORCID:  
<<https://orcid.org/0000-0002-8862-4250>>)

**Maintainer** Jiena McLellan <jienagu90@gmail.com>

**Repository** CRAN

**Date/Publication** 2020-08-13 09:10:06 UTC

## Contents

flashCard . . . . .	2
flashCard-shiny . . . . .	3
<b>Index</b>	<b>4</b>

---

`flashCard`*Create a flash card*

---

**Description**

Create a flash card with desired data frame

**Usage**

```
flashCard(  
  data,  
  frontColor = "#090e87",  
  backColor = "#3443c9",  
  front_text_color = "white",  
  back_text_color = "white",  
  width = "300px",  
  height = "135px",  
  elementId = NULL  
)
```

**Arguments**

<code>data</code>	df with front and back columns
<code>frontColor</code>	frond side background color
<code>backColor</code>	back side background color
<code>front_text_color</code>	frond text color
<code>back_text_color</code>	back text color
<code>width</code>	width
<code>height</code>	height
<code>elementId</code>	element id

**Examples**

```
df1 <- data.frame(  
  front = c("Title front", "contentfront", "content second line"),  
  back = c("Title back", "content back", "second line")  
)  
flashCard(df1, elementId = "card", front_text_color = "grey")
```

---

flashCard-shiny      *Shiny bindings for flashCard*

---

**Description**

Output and render functions for using flashCard within Shiny applications and interactive Rmd documents.

**Usage**

```
flashCardOutput(outputId, width = "300px", height = "135px")  
renderFlashCard(expr, env = parent.frame(), quoted = FALSE)
```

**Arguments**

outputId	output variable to read from
width	width of the widget
height	height of the widget
expr	An expression that generates a flashCard
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.

# Index

flashCard, [2](#)

flashCard-shiny, [3](#)

flashCardOutput (flashCard-shiny), [3](#)

renderFlashCard (flashCard-shiny), [3](#)