

Package ‘shinymaterial’

February 2, 2019

Type Package

Title Implement Material Design in Shiny Applications

Version 0.5.4

Maintainer Eric Anderson <eric.ray.anderson@gmail.com>

Description Allows shiny developers to incorporate UI elements based on Google's Material design. See <<https://material.io/guidelines/>> for more information.

URL <https://ericrayanderson.github.io/shinymaterial/>

License GPL-3 | file LICENSE

Imports shiny (>= 0.7.0), jsonlite

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

material_button	2
material_card	3
material_checkbox	3
material_column	4
material_date_picker	4
material_depth	5
material_dropdown	5
material_file_input	6
material_floating_button	7
material_input	8
material_modal	9
material_number_box	10
material_page	10
material_parallax	11
material_password_box	12
material_radio_button	12
material_row	13
material_side_nav	13
material_side_nav_tabs	14
material_side_nav_tab_content	15
material_slider	15

material_spinner_show	16
material_switch	17
material_tabs	18
material_tab_content	18
material_text_box	19
open_material_modal	19
side_nav_tabs_info	20
update_material_checkbox	20
update_material_date_picker	21
update_material_dropdown	21
update_material_number_box	22
update_material_password_box	23
update_material_radio_button	23
update_material_slider	24
update_material_switch	24
update_material_text_box	25

Index	26
--------------	-----------

material_button	<i>Create a shinymaterial button</i>
-----------------	--------------------------------------

Description

Build a shinymaterial button. The initial value is zero, and increases by one on each press.

Usage

```
material_button(input_id, label, icon = NULL, depth = NULL,
  color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The button text.
icon	String. The name of the icon. Leave empty for no icon. Visit http://materializecss.com/icons.html for a list of available icons.
depth	Integer. The amount of depth of the button. The value should be between 0 and 5. Leave empty for the default depth.
color	String. The color of the button. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors.

Examples

```
material_button(
  input_id = "example_button",
  label = "Button",
  icon = "cloud",
  depth = 5,
  color = "blue lighten-2"
)
```

material_card	Create a card that will contain UI content
---------------	--

Description

UI content can be placed in cards to organize items on a page.

Usage

```
material_card(title, ..., depth = NULL, color = NULL,
              divider = FALSE)
```

Arguments

title	String. The title of the card
...	The UI elements to place in the card
depth	Integer. The amount of depth of the card. The value should be between 0 and 5. Leave empty for the default depth.
color	String. The color of the card background. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors.
divider	logical. Should there be a divider element between card title and card content?

Examples

```
material_card(
  title = "Example Card",
  depth = 5,
  shiny::tags$h5("Card Content")
)
```

material_checkbox	Create a shinymaterial checkbox
-------------------	---------------------------------

Description

Build a shinymaterial checkbox. The value is a boolean (TRUE if checked, FALSE if not checked).

Usage

```
material_checkbox(input_id, label, initial_value = FALSE, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The checkbox label.
initial_value	Boolean. Is the checkbox initially checked?
color	String. The color of the check. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_checkbox(
  input_id = "example_checkbox",
  label = "Checkbox",
  initial_value = TRUE,
  color = "#ef5350"
)
```

material_column	Create a column to organize UI content
-----------------	--

Description

UI content can be placed in columns to organize items on a page.

Usage

```
material_column(..., width = 6, offset = 0)
```

Arguments

...	The UI elements to place in the column.
width	Integer. The width of the column. The value should be between 1 and 12.
offset	Integer. The offset to the left of the column. The value should be between 0 and 11.

Examples

```
material_column(
  width = 4,
  shiny::tags$h1("Column Content")
)
```

material_date_picker	Create a shinymaterial date picker
----------------------	------------------------------------

Description

Build a shinymaterial date picker.

Usage

```
material_date_picker(input_id, label, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The date picker label.
color	String. The date picker color. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors.

Examples

```
material_date_picker(  
  input_id = "example_date_picker",  
  label = "Date Picker"  
)
```

material_depth*Add depth to a UI element*

Description

Give a UI element the perception of depth by creating a shadow.

Usage

```
material_depth(..., depth = 4)
```

Arguments

...	The UI elements to apply the depth.
depth	Integer. The amount of depth. The value should be between 0 and 5. A value of 0 can be used to remove depth from objects that have depth by default.

Examples

```
material_depth(  
  depth = 5,  
  material_card(title = "Example Depth")  
)
```

material_dropdown*Create a shinymaterial dropdown*

Description

Build a shinymaterial dropdown.

Usage

```
material_dropdown(input_id, label, choices = NULL, selected = NULL,  
  multiple = NULL, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The dropdown label.
choices	Named vector. The option names and underlying values.
selected	String. The initially selected underlying value.
multiple	Boolean. Can multiple items be selected?
color	String. The color of the dropdown choices. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_dropdown(
  input_id = "example_dropdown",
  label = "Drop down",
  choices = c(
    "Chicken" = "c",
    "Steak" = "s",
    "Fish" = "f"
  ),
  selected = c("c"),
  multiple = FALSE,
  color = "#ef5350"
)
```

material_file_input *Create a shiny material file input*

Description

Build a shiny material file input.

Usage

```
material_file_input(input_id, label = "File", color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The file input button text.
color	String. The color of the file input. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```

if (interactive()) {

  ui <-
  material_page(
    material_row(
      material_column(
        width = 12,
        material_file_input(
          input_id = "file_1",
          label = "file"
        )
      )
    ),
    material_row(
      material_column(
        width = 12,
        tableOutput("contents")
      )
    )
  )

  server <- function(input, output) {
    output$contents <- renderTable({
      # input$file_1 will be NULL initially. After the user selects
      # and uploads a file, it will be a data frame with 'name',
      # 'size', 'type', and 'datapath' columns. The 'datapath'
      # column will contain the local filenames where the data can
      # be found.
      in_file <- input$file_1

      if (is.null(in_file))
        return(NULL)

      read.csv(in_file$datapath)
    })
  }

  shinyApp(ui, server)
}

```

material_floating_button

Create a shinymaterial floating button

Description

Build a shinymaterial floating button. The initial value is zero, and increases by one on each press.

Usage

```

material_floating_button(input_id, icon = NULL, depth = NULL,
  color = NULL)

```

Arguments

input_id	String. The input identifier used to access the value.
icon	String. The name of the icon. Leave empty for no icon. Visit http://materializecss.com/icons.html for a list of available icons.
depth	Integer. The amount of depth of the floating button. The value should be between 0 and 5. Leave empty for the default depth.
color	String. The color of the floating button. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors.

Examples

```
material_floating_button(
  input_id = "example_floating_button",
  icon = "mode_edit",
  depth = 5,
  color = "red lighten-3"
)
```

material_input	<i>Create a shinymaterial input</i>
----------------	-------------------------------------

Description

Build a shinymaterial input.

Usage

```
material_input(type, ...)
```

Arguments

type	String. The type of input to be created.
...	Additional arguments for the input.

Examples

```
##-- switch --##
material_input(
  type = "switch",
  input_id = "example_switch",
  label = "Switch",
  off_label = "Off",
  on_label = "On",
  initial_value = TRUE
)
```

material_modal	<i>Place UI content in a modal</i>
----------------	------------------------------------

Description

Put any UI object inside of a modal. The modal will open when the button is pressed.

Usage

```
material_modal(modal_id, button_text, title, ..., button_icon = NULL,
               floating_button = FALSE, button_depth = NULL, button_color = NULL,
               close_button_label = "Close", display_button = TRUE)
```

Arguments

modal_id	String. The ID for the modal. Must be unique per application.
button_text	String. The text displayed on the modal trigger button.
title	String. The title of the modal window.
...	The UI elements to place in the modal
button_icon	String. The name of the icon. Visit http://materializecss.com/icons.html for a list of available icons.
floating_button	Boolean. Should the modal trigger button be a floating button?
button_depth	Integer. The amount of depth of the button. The value should be between 0 and 5. Leave empty for the default depth.
button_color	String. The color of the button. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors.
close_button_label	String. The label of the modal close button.
display_button	Boolean. Should the button be displayed in the app? (If FALSE, <code>open_material_modal()</code> may be used to open the modal).

Examples

```
material_modal(
  modal_id = "example_modal",
  button_text = "Modal",
  title = "Example Modal Title",
  button_color = "red lighten-3",
  shiny::tags$p("Modal Content")
)
```

material_number_box	Create a shinymaterial number box
---------------------	-----------------------------------

Description

Build a shinymaterial number box.

Usage

```
material_number_box(input_id, label, min_value, max_value, step_size = 1,
  initial_value, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The number box label.
min_value	Number. The minimum allowable value.
max_value	Number. The maximum allowable value.
step_size	Number. The step size of the arrow clicks.
initial_value	Number. The initial value.
color	String. The accent color of the number box. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_number_box(
  input_id = "example_number_box",
  label = "number box",
  min_value = 5,
  max_value = 15,
  initial_value = 10,
  step_size = 2,
  color = "#ef5350"
)
```

material_page	Create a shinymaterial page
---------------	-----------------------------

Description

Build a shinymaterial page.

Usage

```
material_page(..., title = "", nav_bar_fixed = FALSE,
  nav_bar_color = NULL, background_color = "grey lighten-4",
  font_color = NULL, include_fonts = FALSE, include_nav_bar = TRUE)
```

Arguments

...	The UI elements to place in the page.
title	String. The title of the page.
nav_bar_fixed	Boolean. Should the nav bar remain fixed on the screen?
nav_bar_color	Color of the nav bar. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors.
background_color	Page background color. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors.
font_color	String. The title font color. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>Title color requires using word forms of colors (e.g. "deep-purple"). Also, lighten or darken effects do not work on title colors.</i>
include_fonts	Boolean. Should the material font files be included? (This will place the font sources in a directory 'www', at the same location as the app code.)
include_nav_bar	Boolean. Should the material nav bar be included?

Examples

```
material_page(
  title = "Example Title",
  nav_bar_fixed = TRUE,
  nav_bar_color = "red lighten-2",
  background_color = "blue lighten-4",
  shiny::tags$h1("Page Content")
)
```

material_parallax	Create a parallax image
-------------------	-------------------------

Description

Use this function to create a parallax effect in your application.

Usage

```
material_parallax(image_source)
```

Arguments

image_source	String. The image file name. Place the image in a folder labeled 'www' at the same level as the application (server.R & ui.R).
--------------	--

Examples

```
material_parallax(
  image_source = "example_image.jpg"
)
```

material_password_box *Create a shiny material password box*

Description

Build a shiny material password box.

Usage

```
material_password_box(input_id, label, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The password box label.
color	String. The accent color of the password box. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_password_box(
  input_id = "example_password_box",
  label = "password box",
  color = "#ef5350"
)
```

material_radio_button *Create a shiny material radio button*

Description

Build a shiny material radio button.

Usage

```
material_radio_button(input_id, label, choices, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The radio button label.
choices	Named vector. The option names and underlying values.
color	String. The color of the radio buttons. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_radio_button(  
  input_id = "example_radio_button",  
  label = "Radio Button",  
  choices = c(  
    "Cake" = "c",  
    "Pie" = "p",  
    "Brownie" = "b"  
  ),  
  color = "#ef5350"  
)
```

material_row	Create a row to organize UI content
--------------	-------------------------------------

Description

UI content can be placed in a row to organize items on a page.

Usage

```
material_row(...)
```

Arguments

... The UI elements to place in the row.

Examples

```
material_row(  
  shiny::tags$h1("Row Content")  
)
```

material_side_nav	Create a side-nav that contains UI content
-------------------	--

Description

UI content can be placed in side-nav.

Usage

```
material_side_nav(..., fixed = FALSE, image_source = NULL,  
  background_color = NULL)
```

Arguments

...	The UI elements to place in the side-nav.
fixed	Boolean. Set to TRUE to keep side-nav open on large screens.
image_source	String. The background image file name. Place the image in a folder labeled 'www' at the same level as the application (server.R & ui.R).
background_color	Side-nav background color. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors.

Examples

```
material_side_nav(
  fixed = FALSE,
  image_source = "example_image.jpg",
  background_color = "blue lighten-4",
  shiny::tags$h1("Side-Nav Content")
)
```

material_side_nav_tabs

Place UI content within a side-nav tab

Description

Use this function to create side-nav tabs in your application.

Usage

```
material_side_nav_tabs(side_nav_tabs, icons = NULL, color = NULL,
  font_color = NULL)
```

Arguments

side_nav_tabs	Named vector. The side-nav tab display names and corresponding side-nav tab ids.
icons	String vector. The names of the icons. Leave blank for no icons, or use "none". The length of the vector must match the length of side_nav_tabs. Visit http://materializecss.com/icons.html for a list of available icons.
color	String. The accent color of the side-nav tab wave animation. Leave blank for the default color. Visit http://materializecss.com/waves.html for a list of available colors. <i>Side-nav tab color requires using word forms of colors (e.g. "purple").</i>
font_color	String. The side-nav tabs font color. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>Side-nav tab color requires using word forms of colors (e.g. "deep-purple"). Also, lighten or darken effects do not work on side-nav tab colors.</i>

Examples

```
material_side_nav_tabs(
  side_nav_tabs = c(
    "Example Side-Nav Tab 1" = "example_side_nav_tab_1",
    "Example Side-Nav Tab 2" = "example_side_nav_tab_2"
  ),
  icons = c("cloud", "none"),
  color = "teal"
)
```

```
material_side_nav_tab_content
```

Place UI content within a side-nav tab

Description

Use this function to place UI content within a specific side-nav tab.

Usage

```
material_side_nav_tab_content(side_nav_tab_id, ...)
```

Arguments

`side_nav_tab_id` String. The side-nav tab id in which to place the UI content.

`...` The UI elements to place in the side-nav tab.

Examples

```
material_side_nav_tab_content(
  side_nav_tab_id = "example_side_nav_tab_1",
  shiny::tags$h1("Side-Nav Tab Content")
)
```

```
material_slider
```

Create a shinymaterial slider

Description

Build a shinymaterial slider.

Usage

```
material_slider(input_id, label, min_value, max_value, step_size = 1,
  initial_value, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The slider label.
min_value	Number. The minimum value on the slider.
max_value	Number. The maximum value on the slider.
step_size	Number. The size of step in the slider.
initial_value	Number. The initial value of the slider.
color	String. The slider color. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_slider(
  input_id = "example_slider",
  label = "slider",
  min_value = 5,
  max_value = 15,
  initial_value = 10,
  step_size = 3,
  color = "#ef5350"
)
```

material_spinner_show *Create a material preloader (spinner)*

Description

Display a preloader (spinner) while server is busy.

Usage

```
material_spinner_show(session, output_id)
```

Arguments

session	The session object passed to function given to shinyServer.
output_id	The output id for which the spinner will be a placeholder for.

Examples

```
if(interactive()){
  library(shiny)
  library(shinymaterial)

  ui <- material_page(
    title = "Spinner Example",
    numericInput(inputId = "n", label = "", value = 10),
    plotOutput("n_plot")
  )
}
```



```

)

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

  output$n_plot <- renderPlot({

    #--- Show the spinner ---#
    material_spinner_show(session, "n_plot")

    #--- Simulate calculation step ---#
    Sys.sleep(time = 5)

    #--- Hide the spinner ---#
    material_spinner_hide(session, "n_plot")

    plot(1:input$n)
  })

}

shinyApp(ui = ui, server = server)
}

```

material_switch

Create a shiny material switch

Description

Build a shiny material switch. The value is a boolean (TRUE if 'on', FALSE if 'off').

Usage

```
material_switch(input_id, label, off_label = "", on_label = "",
  initial_value = FALSE, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The switch label.
off_label	String. The label for the 'off' position of the switch.
on_label	String. The label for the 'on' position of the switch.
initial_value	Boolean. Is the switch initially on?
color	String. The color of the switch. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_switch(
  input_id = "example_switch",
  label = "Switch",
  off_label = "Off",
```

```

    on_label = "On",
    initial_value = TRUE,
    color = "#ef5350"
  )

```

material_tabs	<i>Place UI content within a tab</i>
---------------	--------------------------------------

Description

Use this function to create tabs in your application.

Usage

```
material_tabs(tabs, color = NULL)
```

Arguments

tabs	Named vector. The tab display names and corresponding tab ids.
color	String. The accent color of the tabs. Leave blank for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>Tab color requires using word forms of colors (e.g. "deep-purple"). Also, lighten or darken effects do not work on tab colors.</i>

Examples

```

material_tabs(
  tabs = c(
    "Example Tab 1" = "example_tab_1",
    "Example Tab 2" = "example_tab_2"
  ),
  color = "deep-purple"
)

```

material_tab_content	<i>Place UI content within a tab</i>
----------------------	--------------------------------------

Description

Use this function to place UI content within a specific tab.

Usage

```
material_tab_content(tab_id, ...)
```

Arguments

tab_id	String. The tab id in which to place the UI content.
...	The UI elements to place in the tab.

Examples

```
material_tab_content(
  tab_id = "example_tab_1",
  shiny::tags$h1("Tab Content")
)
```

material_text_box	Create a shiny material text box
-------------------	----------------------------------

Description

Build a shiny material text box.

Usage

```
material_text_box(input_id, label, color = NULL)
```

Arguments

input_id	String. The input identifier used to access the value.
label	String. The text box label.
color	String. The accent color of the text box. Leave empty for the default color. Visit http://materializecss.com/color.html for a list of available colors. <i>This input requires using color hex codes, rather than the word form. E.g., "#ef5350", rather than "red lighten-1".</i>

Examples

```
material_text_box(
  input_id = "example_text_box",
  label = "text box",
  color = "#ef5350"
)
```

open_material_modal	Open a material modal programmatically.
---------------------	---

Description

Open a material modal programmatically (server side).

Usage

```
open_material_modal(session, modal_id)
```

Arguments

session	The session object passed to function given to shinyServer.
modal_id	String. The ID of the modal to open.

Examples

```
## Not run:
open_material_modal(session, "example_modal")

## End(Not run)
```

side_nav_tabs_info	<i>Query information about the side-nav tabs.</i>
--------------------	---

Description

Query information about the side-nav tabs (e.g. which tab is active).

Usage

```
side_nav_tabs_info(input = NULL)
```

Arguments

input	The input object in the shiny session.
-------	--

Examples

```
## Not run:
side_nav_tabs_info(input)

## End(Not run)
```

update_material_checkbox	<i>Change the value of a material_checkbox on the client</i>
--------------------------	--

Description

Change the value of a material_checkbox on the client.

Usage

```
update_material_checkbox(session, input_id, value = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_checkbox.
value	Boolean. The value to set for the material_checkbox.

Examples

```
## Not run:
update_material_checkbox(
  session,
  input_id = "example_checkbox",
  value = TRUE
)

## End(Not run)
```

update_material_date_picker

Change the value of a material_date_picker on the client

Description

Change the value of a material_date_picker on the client.

Usage

```
update_material_date_picker(session, input_id, value = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_date_picker.
value	The value to set for the material_date_picker.

Examples

```
## Not run:
update_material_date_picker(
  session,
  input_id = "example_date_picker",
  value = "10 April, 2012"
)

## End(Not run)
```

update_material_dropdown

Change the value of a material_dropdown on the client

Description

Change the value of a material_dropdown on the client.

Usage

```
update_material_dropdown(session, input_id, value = NULL,
  choices = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_dropdown.
value	The value to set for the material_dropdown.
choices	The choices to set for the material_dropdown.

Examples

```
## Not run:
update_material_dropdown(
  session,
  input_id = "example_dropdown",
  value = "New Text"
)

## End(Not run)
```

update_material_number_box

Change the value of a material_number_box on the client

Description

Change the value of a material_number_box on the client.

Usage

```
update_material_number_box(session, input_id, value = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_number_box.
value	The value to set for the material_number_box.

Examples

```
## Not run:
update_material_number_box(
  session,
  input_id = "example_number_box",
  value = 3
)

## End(Not run)
```

`update_material_password_box`*Change the value of a material_password_box on the client*

Description

Change the value of a material_password_box on the client.

Usage

```
update_material_password_box(session, input_id, value = NULL)
```

Arguments

<code>session</code>	The session object passed to function given to shinyServer.
<code>input_id</code>	The input_id of the material_password_box.
<code>value</code>	The value to set for the material_password_box.

Examples

```
## Not run:  
update_material_password_box(  
  session,  
  input_id = "example_password_box",  
  value = "New Password"  
)  
  
## End(Not run)
```

`update_material_radio_button`*Change the value of a material_radio_button on the client*

Description

Change the value of a material_radio_button on the client.

Usage

```
update_material_radio_button(session, input_id, value = NULL)
```

Arguments

<code>session</code>	The session object passed to function given to shinyServer.
<code>input_id</code>	The input_id of the material_radio_button.
<code>value</code>	The value to set for the material_radio_button.

Examples

```
## Not run:
update_material_radio_button(
  session,
  input_id = "example_radio_button",
  value = "new_value"
)

## End(Not run)
```

update_material_slider

Change the value of a material_slider on the client

Description

Change the value of a material_slider on the client.

Usage

```
update_material_slider(session, input_id, value = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_slider.
value	The value to set for the material_slider.

Examples

```
## Not run:
update_material_slider(
  session,
  input_id = "example_slider",
  value = "new_value"
)

## End(Not run)
```

update_material_switch

Change the value of a material_switch on the client

Description

Change the value of a material_switch on the client.

Usage

```
update_material_switch(session, input_id, value = NULL)
```


Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_switch.
value	Boolean. The value to set for the material_switch.

Examples

```
## Not run:
update_material_switch(
  session,
  input_id = "example_switch",
  value = TRUE
)

## End(Not run)
```

update_material_text_box

Change the value of a material_text_box on the client

Description

Change the value of a material_text_box on the client.

Usage

```
update_material_text_box(session, input_id, value = NULL)
```

Arguments

session	The session object passed to function given to shinyServer.
input_id	The input_id of the material_text_box.
value	The value to set for the material_text_box.

Examples

```
## Not run:
update_material_text_box(
  session,
  input_id = "example_text_box",
  value = "New Text"
)

## End(Not run)
```

Index

material_button, [2](#)
material_card, [3](#)
material_checkbox, [3](#)
material_column, [4](#)
material_date_picker, [4](#)
material_depth, [5](#)
material_dropdown, [5](#)
material_file_input, [6](#)
material_floating_button, [7](#)
material_input, [8](#)
material_modal, [9](#)
material_number_box, [10](#)
material_page, [10](#)
material_parallax, [11](#)
material_password_box, [12](#)
material_radio_button, [12](#)
material_row, [13](#)
material_side_nav, [13](#)
material_side_nav_tab_content, [15](#)
material_side_nav_tabs, [14](#)
material_slider, [15](#)
material_spinner_hide
 (material_spinner_show), [16](#)
material_spinner_show, [16](#)
material_switch, [17](#)
material_tab_content, [18](#)
material_tabs, [18](#)
material_text_box, [19](#)

open_material_modal, [19](#)

side_nav_tabs_info, [20](#)

update_material_checkbox, [20](#)
update_material_date_picker, [21](#)
update_material_dropdown, [21](#)
update_material_number_box, [22](#)
update_material_password_box, [23](#)
update_material_radio_button, [23](#)
update_material_slider, [24](#)
update_material_switch, [24](#)
update_material_text_box, [25](#)