Package 'shinybody'

January 8, 2025

Title An Interactive Anatomography Widget for 'shiny'

Version 0.1.3

Description An 'htmlwidget' of the human body that allows you to hide/show and assign colors to 79 different body parts. The 'human' widget is an 'htmlwidget', so it works in Quarto documents, R Markdown documents, or any other HTML medium. It also functions as an input/output widget in a 'shiny' app.

URL https://github.com/robert-norberg/shinybody

BugReports https://github.com/robert-norberg/shinybody/issues

Imports htmlwidgets (>= 0.8), htmltools, crosstalk

Suggests knitr, rmarkdown, shiny

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Depends R (>= 2.10)

LazyData true

NeedsCompilation no

Author Robert Norberg [aut, cre], Sebastian Zapata-Tamayo [aut, ctb], Mehrun Huda [aut, ctb] (<https://orcid.org/0000-0002-4951-8906>), Moffitt Cancer Center [cph]

Maintainer Robert Norberg <Robert.Norberg@moffitt.org>

Repository CRAN

Date/Publication 2025-01-08 14:40:02 UTC

Contents

---

human *Interactive Human Body Widget*

Description

This widget visualizes an SVG-based human body, highlights specific body parts, and displays associated participant data.

Usage

```
human(
  gender = c("male", "female"),
  organ_df,
  select_color = "yellow",
  width = NULL,
  height = NULL,
  elementId = NULL
)
```

Arguments

gender	One of "male" or "female"
organ_df	A data.frame with at least an organ column, and optionally the following columns: • show: A logical (Boolean) column indicating whether or not each organ should be visible. If absent, all organs will be shown. • selected: A logical (Boolean) column indicating whether or not each organ should be in a "selected" state. If absent, no organs will be selected. • hovertext: A character column or a column containing shiny.tag objects. This will be the contents of the tooltip that appears when the organ is hovered over. If absent, the tooltip will contain the title-cased name of the organ (underscores replaced with spaces). • color: A character column indicating the color of the organ. If absent, all organs will be shown in black. If organ_df has other columns, these will be ignored.
select_color	The color that should be applied to organs with the "selected" state (activated by clicking the organ and deactivated by clicking again).
width	Widget width
height	Widget height
elementId	ID of the widget

Value

An object of class human and class htmlwidget.

Examples

```
example_organs <- c("brain", "eye", "heart", "stomach", "bladder")
my_organ_df <- subset(shinybody_organs, organ %in% example_organs)
my_organ_df$show <- TRUE
my_organ_df$color <- grDevices::rainbow(nrow(my_organ_df))
my_organ_df$selected[1] <- TRUE
my_organ_df$hovertext <- mapply(
  function(o, clr) htmltools::strong(
    tools::toTitleCase(o),
    style = paste("color:", clr)
  ),
  my_organ_df$organ,
  my_organ_df$color,
  SIMPLIFY = FALSE
)
human(gender = "female", organ_df = my_organ_df)
```

---

human-shiny *Shiny bindings for human*

Description

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

Usage

```
humanOutput(outputId, width = "100%", height = "400px")

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

Arguments

outputId	output variable to read from
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 human
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

A shiny.tag.list object (in the case of humanOutput) or a shiny.render.function object (in the case of renderHuman).

Examples

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

  male_organs <- shinybody::shinybody_organs$organ[shinybody::shinybody_organs$male]
  female_organs <- shinybody::shinybody_organs$organ[shinybody::shinybody_organs$female]

  ui <- function() {
    fluidPage(
      selectInput(
        inputId = "gender",
        label = "Select Gender",
        choices = c("male", "female"),
        multiple = FALSE,
        selected = "male"
      ),
      selectInput(
        inputId = "body_parts",
        label = "Select Body Parts to Show",
        choices = male_organs,
        multiple = TRUE,
        selected = male_organs[1:5]
      ),
      humanOutput(outputId = "human_widget"),
      verbatimTextOutput(outputId = "clicked_body_part_msg"),
      verbatimTextOutput(outputId = "selected_body_parts_msg")
    )
  }

  server <- function(input, output, session) {
    observe({
      g <- input$gender
      if (g == "male") {
        organ_choices <- male_organs
      } else {
        organ_choices <- female_organs
      }

      updateSelectInput(
        session = session,
        inputId = "body_parts",
        choices = organ_choices,
        selected = organ_choices[1:5]
      )
    })

    output$human_widget <- renderHuman({
      selected_organ_df <- subset(
        shinybody::shinybody_organs,
        organ %in% input$body_parts
      )
      selected_organ_df$show <- TRUE
```

```
      human(
        organ_df = selected_organ_df,
        select_color = "red"
      )
    })
    output$clicked_body_part_msg <- renderPrint({
      paste("You Clicked:", input$clicked_body_part)
    })
    output$selected_body_parts_msg <- renderPrint({
      paste("Selected:", paste(input$selected_body_parts, collapse = ", "))
    })
  }

  shinyApp(ui = ui, server = server)
}
```

`patients` *Example data set of patients*

Description

A randomly generated dataset of patient details

Usage

```
patients
```

Format

`patients`**:**

A data frame with 16 rows and 5 columns:

patient_id A unique patient identifier

gender "male" or "female"

age Patient age

height Patient height in inches

weight Patient weight in lbs

shinybody_organs *Organs available in shinybody*

Description

A list of the organs that shinybody can display

Usage

```
shinybody_organs
```

Format

`shinybody_organs`**:**
A data frame with 79 rows and 7 columns:

organ The name of the organ the row describes (must be unique)

male Boolean. TRUE if the body part can be shown on the male avatar, FALSE otherwise.

female Boolean. TRUE if the body part can be shown on the female avatar, FALSE otherwise.

show Boolean. TRUE if the body part should be shown, FALSE if it should be hidden.

selected Boolean. TRUE if the body part should appear in a "selected" state, FALSE otherwise.

hovertext A character column or a column containing `shiny.tag` objects. This will be the contents of the tooltip that appears when the organ is hovered over. If absent, the tooltip will contain the title-cased name of the organ (underscores replaced with spaces).

color A character column indicating the color the organ should appear if shown.

tumors *Example data set of tumors*

Description

A randomly generated dataset of tumors to use in examples

Usage

```
tumors
```

Format

`tumors`**:**
A data frame with 39 rows and 5 columns:

patient_id A unique patient identifier

tumor_id A unique tumor identifier

tumor_location The organ affected by the tumor

is_primary_tumor TRUE if the tumor is the patient's primary cancer site, otherwise FALSE

stage The stage of the tumor (I, II, or III)

Index