

Package: ROvis.shiny (via r-universe)

October 19, 2026

Title Create interactive components for interactive data visualisations in Rijksoverheid style using 'shiny'

Version 0.1.0

Description Provides a comprehensive suite of utilities for interactive visualisation in Rijksoverheid style.

License EUPL (≥ 1.2)

Imports cli (>= 3.6.2), dplyr (>= 1.1.4), DT (>= 0.34.0), htmltools (>= 0.5.8.1), magrittr (>= 2.0.3), rlang (>= 1.1.0), ROvis.table, ROvis.plotly, shiny (>= 1.11.1), shinyjs (>= 2.1.1), utils

Suggests covr (>= 3.6.4), pkgdown (>= 2.0.9), quarto (>= 1.4),
testthat (>= 3.0.0), ggplot2, highcharter (>= 0.9.4), echarts4r
(>= 0.4.6), plotly (>= 4.10.4), tibble, tidyr, webshot2,
ROvis.utils, knitr, rmarkdown

Depends R ($\geq 4.5.0$)

Remotes rivm-syso/ROvis.utils, rivm-syso/ROvis.table,
rivm-syso/ROvis.plotly

VignetteBuilder quarto

Config/roxygen2/version 8.1.0

Config/testthat/edition 3

Encoding UTF-8

Config/roxygen2/markdown TRUE

```
Config/pak/sysreqs cmake libfontconfig1-dev libfreetype6-dev make libicu-dev libuv1-dev libxml2-  
dev libssl-dev libnode-dev zlib1g-dev
```

Repository <https://rivm-syso.r-universe.dev>

Date/Publication 2026-10-13 11:52:07 UTC

RemoteUrl <https://github.com/rivm-syso/ROvis.shiny>

RemoteRef HEAD

RemoteSha 079c46c4d60f695b6e27f3086881b77069a0af7f

Contents

ro_create_custom_button 2

ro_icon_download 3

ro_icon_table 4

ro_shiny_button_theme 5

ro_shiny_download_server 6

ro_shiny_download_ui 7

ro_shiny_dt_responsive 8

ro_shiny_dt_server 9

ro_shiny_dt_ui 10

ro_shiny_focus_style 10

ro_shiny_graph_mod_server 11

ro_shiny_graph_mod_ui 12

ro_shiny_graph_panel_server 13

ro_shiny_graph_panel_ui 15

ro_shiny_screenreader 16

ro_shiny_skip_link 17

ro_show_table_server 18

ro_show_table_ui 19

Index 20

ro_create_custom_button	
	<i>Create a Custom Button for Shiny UI</i>

Description

Generates a Shiny action button with an SVG icon and custom label text.

Usage

```
ro_create_custom_button(input_id, label_text, svg_path)
```

Arguments

input_id	Character. The input ID for the action button.
label_text	Character. The label to display next to the icon.
svg_path	Character. SVG path data for the icon.

Value

A Shiny tag list representing the action button UI element.

See Also

[ro_icon_download](#), [ro_icon_table](#)

Other DT: [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenerreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
## Not run:
ro_create_custom_button(
  input_id = "download_btn",
  label_text = "Download",
  svg_path = ro_icon_download()
)

## End(Not run)
```

ro_icon_download	<i>SVG path for download icon</i>
------------------	-----------------------------------

Description

SVG path for download icon

Usage

```
ro_icon_download()
```

Value

A character string with the SVG path data for a download icon.

See Also

[ro_create_custom_button](#), [ro_icon_table](#), [ro_shiny_graph_panel_server](#), [ro_shiny_graph_panel_ui](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenerreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
## Not run:
download_button <- ro_icon_download()

## End(Not run)
```

ro_icon_table	<i>SVG path for table icon</i>
---------------	--------------------------------

Description

SVG path for table icon

SVG path for table icon

Usage

```
ro_icon_table()
```

```
ro_icon_table()
```

Value

A character string with the SVG path data for a table icon.

A character string with the SVG path data for a table icon.

See Also

[ro_create_custom_button](#), [ro_icon_download](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
## Not run:  
button_table <- ro_icon_table()  
  
## End(Not run)  
## Not run:  
icon_table <- ro_icon_table()  
  
## End(Not run)
```

ro_shiny_button_theme *Button Theme CSS for Shiny Apps (RIVM Style)*

Description

Provides RIVM house style CSS for custom icon buttons in Shiny apps. Includes styling for button appearance, hover/active/focus states, icon styling, and button container layout.

Usage

```
ro_shiny_button_theme()
```

Value

A shiny.tags object (HTML tag) to be included at the top of your Shiny UI.

See Also

[ro_shiny_skip_link](#), [ro_shiny_skip_link_style](#), [ro_shiny_focus_style](#), [ro_shiny_dt_responsive](#), [ro_create_custom_button](#), [ro_icon_download](#), [ro_icon_table](#)

Other shiny: [ro_shiny_dt_responsive\(\)](#), [ro_shiny_focus_style\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_shiny_skip_link\(\)](#), [ro_shiny_skip_link_style\(\)](#)

Examples

```
if (interactive()) {  
  ui <- fluidPage(  
    ro_shiny_screenreader(),  
    ro_shiny_skip_link(),  
    ro_shiny_skip_link_style(),  
    ro_shiny_focus_style(),  
    ro_shiny_button_theme(),  
    ro_shiny_dt_responsive(),  
    # ... your app UI ...  
  )  
  server <- function(input, output, session) {}  
  shinyApp(ui, server)  
}
```

ro_shiny_download_server

Download Button Module Server

Description

Server logic for a download button that exports data as a CSV file using a custom filename.

Usage

```
ro_shiny_download_server(
  id,
  data_to_download,
  filename_prefix = "data",
  svg_path
)
```

Arguments

id	Character. The module namespace ID.
data_to_download	Function. A function returning the data to be downloaded (as a data frame or tibble).
filename_prefix	Character. Prefix for the downloaded file name. Default is "data".
svg_path	Character. SVG path data for the icon (not used in server, but included for consistency).

Value

None. Sets up the server logic for the download button module.

See Also

[ro_shiny_download_ui](#), [ro_shiny_graph_panel_server](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
# In server function:
if (interactive()) {
  ro_shiny_download_server(
    id = "download1",
    data_to_download = function() mydata,
    filename_prefix = "cases",
```

```
    svg_path = ro_icon_download()  
  )  
}
```

ro_shiny_download_ui *Download Button Module UI*

Description

Creates a Shiny UI element for a download button with a custom SVG icon and label.

Usage

```
ro_shiny_download_ui(id, label_text = "Export", svg_path)
```

Arguments

id	Character. The module namespace ID.
label_text	Character. The label to display on the button. Default is "Export".
svg_path	Character. SVG path data for the icon.

Value

A Shiny UI element (action button) for use in a module UI.

See Also

[ro_shiny_download_server](#), [ro_shiny_graph_panel_ui](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
if (interactive()) {  
  ro_shiny_download_ui(  
    id = "download1",  
    label_text = "Export",  
    svg_path = ro_icon_download()  
  )  
}
```

`ro_shiny_dt_responsive`*Responsive DataTables CSS for Shiny Apps*

Description

Provides CSS to make DataTables responsive in Shiny apps. Enables horizontal scrolling for tables on small screens, ensuring tables remain usable on mobile devices and narrow viewports.

Usage

```
ro_shiny_dt_responsive()
```

Value

A shiny.tags object (HTML tag) to be included at the top of your Shiny UI.

See Also

[ro_shiny_skip_link](#), [ro_shiny_skip_link_style](#), [ro_shiny_focus_style](#), [ro_shiny_button_theme](#), [ro_shiny_dt_ui](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Other shiny: [ro_shiny_button_theme\(\)](#), [ro_shiny_focus_style\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_shiny_skip_link\(\)](#), [ro_shiny_skip_link_style\(\)](#)

Examples

```
if (interactive()) {  
  ui <- fluidPage(  
    ro_shiny_screenreader(),  
    ro_shiny_skip_link(),  
    ro_shiny_skip_link_style(),  
    ro_shiny_focus_style(),  
    ro_shiny_button_theme(),  
    ro_shiny_dt_responsive(),  
    # ... your app UI ...  
  )  
  server <- function(input, output, session) {}  
  shinyApp(ui, server)  
}
```

ro_shiny_dt_server	<i>DT Table Module Server</i>
--------------------	-------------------------------

Description

Server logic for rendering an accessible, keyboard-navigable (with customJS) DT table with options for sorting and page length.

Usage

```
ro_shiny_dt_server(id, data, caption, sorting = FALSE, pagelength = 10)
```

Arguments

id	Character. The module namespace ID.
data	Data frame or tibble. The data to display in the table.
caption	Character. Table caption/title (displayed above the table).
sorting	Logical. Should column sorting be enabled? Default is FALSE.
pagelength	Integer. Number of rows to display per page. Default is 10.

Value

None. Sets up the server logic for the DT table module.

See Also

[ro_shiny_dt_ui](#), [ro_shiny_graph_panel_server](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
# In your Shiny server:
if (interactive()) {
  ro_shiny_dt_server(
    id = "table1",
    data = my_data,
    caption = "Aantal gevallen per groep",
    sorting = TRUE,
    pagelength = 20
  )
}
```

ro_shiny_dt_ui	<i>DT Table Module UI</i>
----------------	---------------------------

Description

Creates a Shiny UI element for displaying a DT (DataTable) table output within a module.

Usage

```
ro_shiny_dt_ui(id)
```

Arguments

id Character. The module namespace ID.

Value

A Shiny UI output element for rendering a DT table.

See Also

[ro_shiny_dt_server](#), [ro_shiny_graph_panel_server](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenerreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
if (interactive()) {
  ro_shiny_dt_ui("table1")
}
```

ro_shiny_focus_style	<i>Focus Indicators CSS for Shiny Apps</i>
----------------------	--

Description

Provides CSS for focus indicators on all interactive elements in Shiny apps. Ensures keyboard users can see which element currently has focus, improving accessibility and keyboard navigation.

Usage

```
ro_shiny_focus_style()
```

Value

A shiny.tags object (HTML tag) to be included at the top of your Shiny UI.

See Also

[ro_shiny_skip_link](#), [ro_shiny_skip_link_style](#), [ro_shiny_button_theme](#), [ro_shiny_dt_responsive](#)

Other shiny: [ro_shiny_button_theme\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_screener\(\)](#), [ro_shiny_skip_link\(\)](#), [ro_shiny_skip_link_style\(\)](#)

Examples

```
if (interactive()) {
  ui <- fluidPage(
    ro_shiny_screener(),
    ro_shiny_skip_link(),
    ro_shiny_skip_link_style(),
    ro_shiny_focus_style(),
    ro_shiny_button_theme(),
    ro_shiny_dt_responsive(),
    # ... your app UI ...
  )
  server <- function(input, output, session) {}
  shinyApp(ui, server)
}
```

ro_shiny_graph_mod_server

Generic Graph Module Server

Description

Server logic for a generic chart module. Renders a chart using a user-supplied render function and a plot-producing function.

Usage

```
ro_shiny_graph_mod_server(id, render_fun, plot_data_fun)
```

Arguments

id	Character. Module namespace ID.
render_fun	Function. A Shiny render function (<code>renderPlotly</code> , <code>highcharter::renderHighchart</code> , <code>echarts4r::renderEcharts4r</code> , etc.).
plot_data_fun	Function. A function returning the data or object to plot.

Value

None. Sets up the server logic for the generic chart module.

See Also

[ro_shiny_graph_panel_server](#), [ro_shiny_graph_mod_ui](#)

Other ggplotly: [ro_shiny_graph_mod_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#)

Other plotly: [ro_shiny_graph_mod_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#)

Other echarts: [ro_shiny_graph_panel_server\(\)](#)

Examples

```
# Using plotly
if (interactive()) {
  ro_shiny_graph_mod_server(
    id = "my_chart",
    render_fun = plotly::renderPlotly,
    plot_data_fun = function() plotly_obj
  )
}
# Using highcharter
if (interactive()) {
  ro_shiny_graph_mod_server(
    id = "my_chart",
    render_fun = highcharter::renderHighchart,
    plot_data_fun = function() highcharter::highchart() |>
      highcharter::hc_add_series(data = c(1,2,3))
  )
}

# Using echarts4r
if (interactive()) {
  ro_shiny_graph_mod_server(
    id = "my_chart",
    render_fun = echarts4r::renderEcharts4r,
    plot_data_fun = function() echarts4r::e_charts(1:3) |> echarts4r::e_line(1:3)
  )
}
```

`ro_shiny_graph_mod_ui` *Generic yGraph Module UI*

Description

Returns a UI output element for a generic chart module, using a user-supplied chart UI function.

Usage

```
ro_shiny_graph_mod_ui(id, plot_ui_fun)
```

Arguments

id	Character. Module namespace ID.
plot_ui_fun	Function. A function that generates a UI element for the chart output (plotlyOutput, highcharter::highchartOutput, echarts4r::echarts4rOutput, etc.).

Value

A Shiny UI output element for use in a module UI.

See Also

[ro_shiny_graph_panel_ui](#), [ro_shiny_graph_mod_server](#)

Other ggplotly: [ro_shiny_graph_mod_server\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#)

Other plotly: [ro_shiny_graph_mod_server\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#)

Other echarts4r: [ro_shiny_graph_panel_ui\(\)](#)

Examples

```
# Using plotly
if (interactive()) {
  ro_shiny_graph_mod_ui("my_chart", plotlyOutput)
}

# Using highcharter
if (interactive()) {
  ro_shiny_graph_mod_ui("my_chart", highcharter::highchartOutput)
}

# Using echarts4r
if (interactive()) {
  ro_shiny_graph_mod_ui("my_chart", echarts4r::echarts4rOutput)
}
```

ro_shiny_graph_panel_server

Graph Panel Module Server

Description

Server logic for the graph panel module. Handles rendering of the chart, toggling and displaying the table, and downloading data.

Usage

```
ro_shiny_graph_panel_server(
  id,
  plot_render_fun,
  plot_data_fun,
  data,
  caption,
  sorting = FALSE,
  pagelength = 10
)
```

Arguments

<code>id</code>	Character. Module namespace ID.
<code>plot_render_fun</code>	Function. A Shiny render function (<code>plotly::renderPlotly</code> , <code>highcharter::renderHighchart</code> , etc.).
<code>plot_data_fun</code>	Function. A function returning the data or plot object for the chart.
<code>data</code>	Data frame or tibble. The data to be shown in the table and available for download.
<code>caption</code>	Character. Caption/title for the data table.
<code>sorting</code>	Logical. Should the table allow column sorting? Default is FALSE. When used, apply the same in <code>ro_shiny_graph_panel_ui</code> .
<code>pagelength</code>	Integer. Number of rows per page in the data table. Default is 10.

Value

None. Sets up the server logic for the graph panel module.

See Also

`ro_shiny_graph_panel_ui`, `ro_shiny_graph_mod_server`, `ro_show_table_server`, `ro_shiny_download_server`, `ro_icon_download`, `ro_shiny_dt_ui`, `ro_shiny_dt_server`

Other ggplotly: `ro_shiny_graph_mod_server()`, `ro_shiny_graph_mod_ui()`, `ro_shiny_graph_panel_ui()`

Other plotly: `ro_shiny_graph_mod_server()`, `ro_shiny_graph_mod_ui()`, `ro_shiny_graph_panel_ui()`

Other echarts: `ro_shiny_graph_mod_server()`

Other DT: `ro_create_custom_button()`, `ro_icon_download()`, `ro_icon_table()`, `ro_shiny_download_server()`, `ro_shiny_download_ui()`, `ro_shiny_dt_responsive()`, `ro_shiny_dt_server()`, `ro_shiny_dt_ui()`, `ro_shiny_graph_panel_ui()`, `ro_shiny_screenreader()`, `ro_show_table_server()`, `ro_show_table_ui()`

Examples

```
# In your Shiny server:
if (interactive()) {
  ro_shiny_graph_panel_server(
    id = "main",
```

```

    plot_render_fun = plotly::renderPlotly,
    plot_data_fun = function() plotly_obj,
    data = my_data,
    caption = "Aantal gevallen per groep",
    sorting = TRUE,
    pagelength = 20
  )
}

```

ro_shiny_graph_panel_ui

Graph Panel Module UI

Description

Generates a Shiny UI module including a chart, skip-link for accessibility, a table toggle button, download button, and a data table area.

Usage

```
ro_shiny_graph_panel_ui(id, plot_ui_fun, sorting = FALSE)
```

Arguments

id	Character. Module namespace ID.
plot_ui_fun	Function. UI function to create the chart output (for example plotlyOutput).
sorting	Logical. Should the table allow column sorting? Default is FALSE. When used, apply the same in ro_shiny_graph_panel_server.

Value

A Shiny UI element (tagList) for use in a module UI.

See Also

[ro_shiny_graph_panel_server](#), [ro_shiny_skip_link](#), [ro_shiny_skip_link_style](#), [ro_shiny_focus_style](#), [ro_shiny_button_theme](#), [ro_shiny_dt_responsive](#), [ro_shiny_screenreader](#), [ro_icon_download](#), [ro_shiny_graph_mod_ui](#), [ro_show_table_ui](#), [ro_shiny_download_ui](#)

Other ggplotly: [ro_shiny_graph_mod_server\(\)](#), [ro_shiny_graph_mod_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#)

Other plotly: [ro_shiny_graph_mod_server\(\)](#), [ro_shiny_graph_mod_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#)

Other echarts4r: [ro_shiny_graph_mod_ui\(\)](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
if (interactive()) {
  ro_shiny_graph_panel_ui("main", plotlyOutput)
}
```

ro_shiny_screenreader *Screen Reader Only CSS Classes for Shiny Apps*

Description

Provides general-purpose CSS classes that visually hide content while keeping it accessible to screen readers. Includes `.sr-only` and `.visually-hidden` classes.

Usage

```
ro_shiny_screenreader()
```

Value

A shiny.tags object with a <head> tag containing screen reader only CSS classes.

See Also

[ro_shiny_skip_link](#), [ro_shiny_skip_link_style](#), [ro_shiny_dt_ui](#), [ro_shiny_graph_panel_ui](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_show_table_server\(\)](#), [ro_show_table_ui\(\)](#)

Other shiny: [ro_shiny_button_theme\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_focus_style\(\)](#), [ro_shiny_skip_link\(\)](#), [ro_shiny_skip_link_style\(\)](#)

Examples

```
if (interactive()) {
  ui <- fluidPage(
    ro_dt_theme(sorting = FALSE),
    ro_shiny_screenreader(),
    DTOutput("mytable")
  )
  server <- function(input, output, session) {
    output$mytable <- renderDT(datatable(iris, options(list(ordering=FALSE))))
  }
  shinyApp(ui, server)
}
```

ro_shiny_skip_link	<i>Skip Link JavaScript Handlers for Shiny Apps</i>
--------------------	---

Description

Provides JavaScript handlers for skip links in Shiny apps. Skip links allow keyboard users to jump directly to the data table, bypassing the chart. Includes download CSV handler and skip-link navigation logic.

Usage

```
ro_shiny_skip_link()
```

Details

This function provides JavaScript functionality only. Use with [ro_shiny_screenreader](#) (for `.sr-only` base class) and [ro_shiny_skip_link_style](#) (for `.sr-only-focusable` styling).

Value

A `shiny.tags` object (HTML tag) to be included at the top of your Shiny UI.

See Also

[ro_shiny_skip_link_style](#), [ro_shiny_focus_style](#), [ro_shiny_button_theme](#), [ro_shiny_dt_responsive](#), [ro_shiny_graph_panel_ui](#), [ro_shiny_screenreader](#)

Other shiny: [ro_shiny_button_theme\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_focus_style\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_shiny_skip_link_style\(\)](#)

Examples

```
if (interactive()) {
  ui <- fluidPage(
    ro_shiny_screenreader(),
    ro_shiny_skip_link(),
    ro_shiny_skip_link_style(),
    ro_shiny_focus_style(),
    ro_shiny_button_theme(),
    ro_shiny_dt_responsive(),
    # ... your app UI ...
  )
  server <- function(input, output, session) {}
  shinyApp(ui, server)
}
```

ro_show_table_server *Table Toggle Button Module Server*

Description

Server logic for toggling the table visibility button and updating its label/icon.

Usage

```
ro_show_table_server(
  id,
  svg_path,
  label_show = "Toon tabel",
  label_hide = "Verberg tabel"
)
```

Arguments

id	Character. The module namespace ID.
svg_path	Character. SVG path data for the icon.
label_show	Character. Label shown when table is hidden. Default is "Toon tabel".
label_hide	Character. Label shown when table is visible. Default is "Verberg tabel".

Value

A list with a reactiveVal show_table, indicating whether the table is shown.

See Also

[ro_show_table_ui](#), [ro_shiny_graph_panel_server](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_ui\(\)](#)

Examples

```
if (interactive()) {
  ro_show_table_server(
    id = "toggle1",
    svg_path = ro_icon_table()
  )
}
```

ro_show_table_ui	<i>Table Toggle Button Module UI</i>
------------------	--------------------------------------

Description

Creates a Shiny action button with an SVG icon to toggle table visibility.

Usage

```
ro_show_table_ui(
  id,
  label_show = "Toon tabel",
  label_hide = "Verberg tabel",
  svg_path
)
```

Arguments

id	Character. The module namespace ID.
label_show	Character. Label shown when table is hidden. Default is "Toon tabel".
label_hide	Character. Label shown when table is visible. Default is "Verberg tabel".
svg_path	Character. SVG path data for the icon.

Value

A Shiny UI element (action button) for use in a module UI.

See Also

[ro_show_table_server](#), [ro_shiny_graph_panel_ui](#)

Other DT: [ro_create_custom_button\(\)](#), [ro_icon_download\(\)](#), [ro_icon_table\(\)](#), [ro_shiny_download_server\(\)](#), [ro_shiny_download_ui\(\)](#), [ro_shiny_dt_responsive\(\)](#), [ro_shiny_dt_server\(\)](#), [ro_shiny_dt_ui\(\)](#), [ro_shiny_graph_panel_server\(\)](#), [ro_shiny_graph_panel_ui\(\)](#), [ro_shiny_screenreader\(\)](#), [ro_show_table_server\(\)](#)

Examples

```
if (interactive()) {
  ro_show_table_ui(
    id = "toggle1",
    svg_path = ro_icon_table()
  )
}
```

Index

* DT

- ro_create_custom_button, 2
- ro_icon_download, 3
- ro_icon_table, 4
- ro_shiny_download_server, 6
- ro_shiny_download_ui, 7
- ro_shiny_dt_responsive, 8
- ro_shiny_dt_server, 9
- ro_shiny_dt_ui, 10
- ro_shiny_graph_panel_server, 13
- ro_shiny_graph_panel_ui, 15
- ro_shiny_screenreader, 16
- ro_show_table_server, 18
- ro_show_table_ui, 19

* echarts4r

- ro_shiny_graph_mod_ui, 12
- ro_shiny_graph_panel_ui, 15

* echarts

- ro_shiny_graph_mod_server, 11
- ro_shiny_graph_panel_server, 13

* ggplotly

- ro_shiny_graph_mod_server, 11
- ro_shiny_graph_mod_ui, 12
- ro_shiny_graph_panel_server, 13
- ro_shiny_graph_panel_ui, 15

* plotly

- ro_shiny_graph_mod_server, 11
- ro_shiny_graph_mod_ui, 12
- ro_shiny_graph_panel_server, 13
- ro_shiny_graph_panel_ui, 15

* shiny

- ro_shiny_button_theme, 5
- ro_shiny_dt_responsive, 8
- ro_shiny_focus_style, 10
- ro_shiny_screenreader, 16
- ro_shiny_skip_link, 17

- ro_create_custom_button, 2, 3–5
- ro_create_custom_button(), 3, 4, 6–10, 14–16, 18, 19

- ro_icon_download, 3, 3–5, 14, 15
- ro_icon_download(), 3, 4, 6–10, 14–16, 18, 19
- ro_icon_table, 3, 4, 5
- ro_icon_table(), 3, 6–10, 14–16, 18, 19
- ro_shiny_button_theme, 5, 8, 11, 15, 17
- ro_shiny_button_theme(), 8, 11, 16, 17
- ro_shiny_download_server, 6, 7, 14
- ro_shiny_download_server(), 3, 4, 7–10, 14–16, 18, 19
- ro_shiny_download_ui, 6, 7, 15
- ro_shiny_download_ui(), 3, 4, 6, 8–10, 14–16, 18, 19
- ro_shiny_dt_responsive, 5, 8, 11, 15, 17
- ro_shiny_dt_responsive(), 3–7, 9–11, 14–19
- ro_shiny_dt_server, 9, 10, 14
- ro_shiny_dt_server(), 3, 4, 6–8, 10, 14–16, 18, 19
- ro_shiny_dt_ui, 8, 9, 10, 14, 16
- ro_shiny_dt_ui(), 3, 4, 6–9, 14–16, 18, 19
- ro_shiny_focus_style, 5, 8, 10, 15, 17
- ro_shiny_focus_style(), 5, 8, 16, 17
- ro_shiny_graph_mod_server, 11, 13, 14
- ro_shiny_graph_mod_server(), 13–15
- ro_shiny_graph_mod_ui, 12, 12, 15
- ro_shiny_graph_mod_ui(), 12, 14, 15
- ro_shiny_graph_panel_server, 3, 6, 9, 10, 12, 13, 15, 18
- ro_shiny_graph_panel_server(), 3, 4, 6–10, 12, 13, 15, 16, 18, 19
- ro_shiny_graph_panel_ui, 3, 7, 13, 14, 15, 16, 17, 19
- ro_shiny_graph_panel_ui(), 3, 4, 6–10, 12–14, 16, 18, 19
- ro_shiny_screenreader, 15, 16, 17
- ro_shiny_screenreader(), 3–11, 14, 15, 17–19
- ro_shiny_skip_link, 5, 8, 11, 15, 16, 17

`ro_shiny_skip_link()`, [5](#), [8](#), [11](#), [16](#)
`ro_shiny_skip_link_style`, [5](#), [8](#), [11](#), [15–17](#)
`ro_shiny_skip_link_style()`, [5](#), [8](#), [11](#), [16](#),
[17](#)
`ro_show_table_server`, [14](#), [18](#), [19](#)
`ro_show_table_server()`, [3](#), [4](#), [6–10](#), [14–16](#),
[19](#)
`ro_show_table_ui`, [15](#), [18](#), [19](#)
`ro_show_table_ui()`, [3](#), [4](#), [6–10](#), [14–16](#), [18](#)