

# Package ‘shinyanimate’

May 9, 2026

**Type** Package

**Title** Animation for 'shiny' Elements

**Version** 0.4.0

**Author** Swechhya Bista

**Maintainer** Swechhya Bista <swechhyabista@gmail.com>

**Description** An extension of 'animate.css' that allows user to easily add animations to any UI element in 'shiny' app using the elements id.

**Imports** shiny

**License** MIT + file LICENSE

**URL** <https://github.com/Swechhya/shinyanimate>

**BugReports** <https://github.com/Swechhya/shinyanimate/issues>

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-10-30 11:30:02 UTC

## Contents

addHoverAnim . . . . .	2
addScrollAnim . . . . .	2
startAnim . . . . .	3
withAnim . . . . .	4
<b>Index</b>	<b>5</b>

---

addHoverAnim	<i>Add animation on mouse hover for UI element.</i>
--------------	---

---

**Description**

Add animation on mouse hover for UI element.

**Usage**

```
addHoverAnim(session, id, type = NULL)
```

**Arguments**

session	The session object passed to function given to shinyServer.
id	the id of the UI element for which you want animation on mouse hover.
type	The type of animation to use, valid values correspond to the types in <a href="https://dandedn.github.io/animate.css/">https://dandedn.github.io/animate.css/</a>

**See Also**

[withAnim](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(shinyanimate)
  ui <- fluidPage(
    withAnim(),
    tags$div(id = 'title', h1('HOVER ON ME'))
  )
  server <- function(input, output, session){
    observe(addHoverAnim(session, 'title', 'bounce'))
  }
  shinyApp(ui, server)
}
```

---

addScrollAnim	<i>Add animation on scroll for UI element.</i>
---------------	--

---

**Description**

Add animation on scroll for UI element.

**Usage**

```
addScrollAnim(session, id, type = NULL)
```

**Arguments**

session	The session object passed to function given to shinyServer.
id	the id of the UI element for which you want animation on scroll.
type	The type of animation to use, valid values correspond to the types in <a href="https://daneden.github.io/animate.css/">https://daneden.github.io/animate.css/</a>

**See Also**

[withAnim](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(shinyanimate)
  ui <- fluidPage(
    withAnim(),
    tags$h1('Scroll below to see an animation'),
    br(), br(), br(), br(), br(), br(), br(), br(),
    br(), br(), br(), br(), br(), br(), br(), br(),
    br(), br(), br(), br(), br(), br(), br(), br(),
    br(), br(), br(), br(), br(), br(), br(), br(),
    br(), br(), br(), br(), br(), br(), br(), br(),
    tags$div(id = 'title', h1('I ANIMATE ON SCROLL'))
  )
  server <- function(input, output, session){
    observe(addScrollAnim(session, 'title', 'bounce'))
  }
  shinyApp(ui, server)
}
```

---

startAnim

*Start an animation*

---

**Description**

Start an animation of the UI element.

**Usage**

```
startAnim(session, id, type = NULL, delay = NULL)
```

**Arguments**

session	The session object passed to function given to shinyServer.
id	the id of the UI element for which you want to add animation.
type	The type of animation to use, valid values correspond to the types in <a href="https://dandedn.github.io/animate.css/">https://dandedn.github.io/animate.css/</a>
delay	The time after which you want to add animation in milliseconds

**See Also**

[withAnim](#)

**Examples**

```
if(interactive()){
  library(shiny)
  library(shinyanimate)
  ui <- fluidPage(
    withAnim(),
    tags$div(id = 'title', h1('ANIMATION')),
    actionButton(inputId = "button", label = "Animate")
  )
  server <- function(input, output, session){
    observeEvent(input$button,{
      startAnim(session, 'title', 'bounce')
    })
  }
  shinyApp(ui, server)
}
```

---

withAnim

*Set up Shiny app to use animation*

---

**Description**

This function needs to be added in the UI if to want to add animation to your UI elements using shinyanimate.

**Usage**

```
withAnim()
```

**See Also**

[startAnim](#)

**Examples**

```
if(interactive()){withAnim()}
```

# Index

`addHoverAnim`, [2](#)  
`addScrollAnim`, [2](#)  
`startAnim`, [3](#), [4](#)  
`withAnim`, [2–4](#), [4](#)