This tutorial is deprecated. Learn more about Shiny at our new location, shiny.rstudio.com.
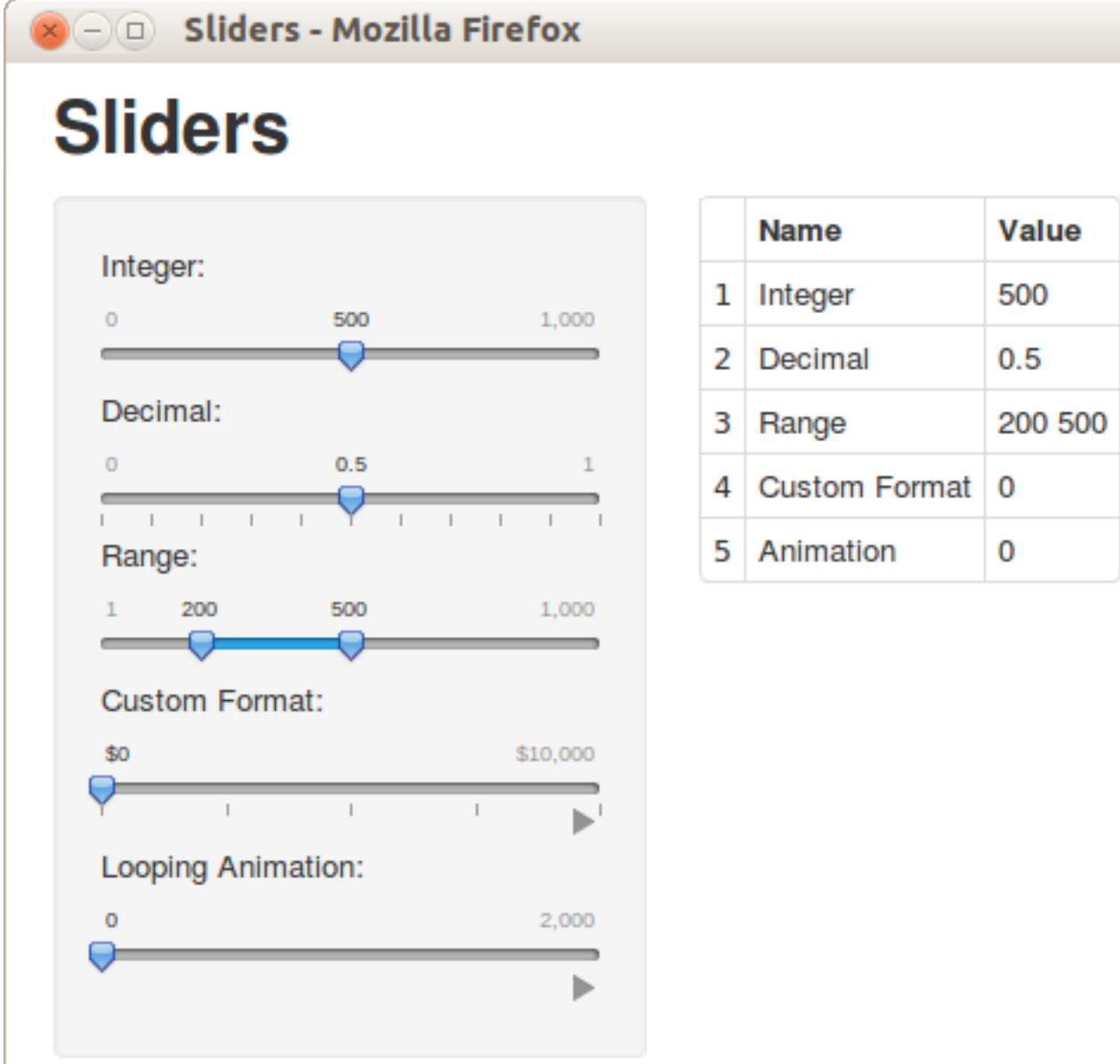
The Sliders application demonstrates the many capabilities of slider controls, including the ability to run an animation sequence. To run the example type:

```
> library(shiny)
> runExample("05_sliders")
```

Customizing Sliders

Shiny slider controls are extremely capable and customizable. Features supported include:

* The ability to input both single values and ranges
* Custom formats for value display (e.g for currency)
* The ability to animate the slider across a range of values

Slider controls are created by calling the `sliderInput` function. The ui.R file demonstrates using sliders with a variety of options:

ui.R

```r
library(shiny)

# Define UI for slider demo application
shinyUI(pageWithSidebar(

  #  Application title
  headerPanel("Sliders"),

  # Sidebar with sliders that demonstrate various available options
  sidebarPanel(
    # Simple integer interval
    sliderInput("integer", "Integer:",
                min=0, max=1000, value=500),

    # Decimal interval with step value
    sliderInput("decimal", "Decimal:",
                min = 0, max = 1, value = 0.5, step= 0.1),

    # Specification of range within an interval
    sliderInput("range", "Range:",
                min = 1, max = 1000, value = c(200,500)),

    # Provide a custom currency format for value display, with basic animation
    sliderInput("format", "Custom Format:",
                min = 0, max = 10000, value = 0, step = 2500,
                format="$#,##0", locale="us", animate=TRUE),

    # Animation with custom interval (in ms) to control speed, plus looping
    sliderInput("animation", "Looping Animation:", 1, 2000, 1, step = 10,
                animate=animationOptions(interval=300, loop=T))
  ),

  # Show a table summarizing the values entered
  mainPanel(
    tableOutput("values")
  )
))
```

Server Script

The server side of the Slider application is very straightforward: it creates a data frame containing all of the input values and then renders it as an HTML table:

server.R

```r
library(shiny)

# Define server logic for slider examples
shinyServer(function(input, output) {

  # Reactive expression to compose a data frame containing all of the values
  sliderValues <- reactive({

    # Compose data frame
    data.frame(
      Name = c("Integer",
               "Decimal",
               "Range",
               "Custom Format",
               "Animation"),
      Value = as.character(c(input$integer,
                             input$decimal,
                             paste(input$range, collapse=' '),
                             input$format,
                             input$animation)),
      stringsAsFactors=FALSE)
  })

  # Show the values using an HTML table
  output$values <- renderTable({
    sliderValues()
  })
})
```

← Previous   Next →