Shiny

Justine Guégan

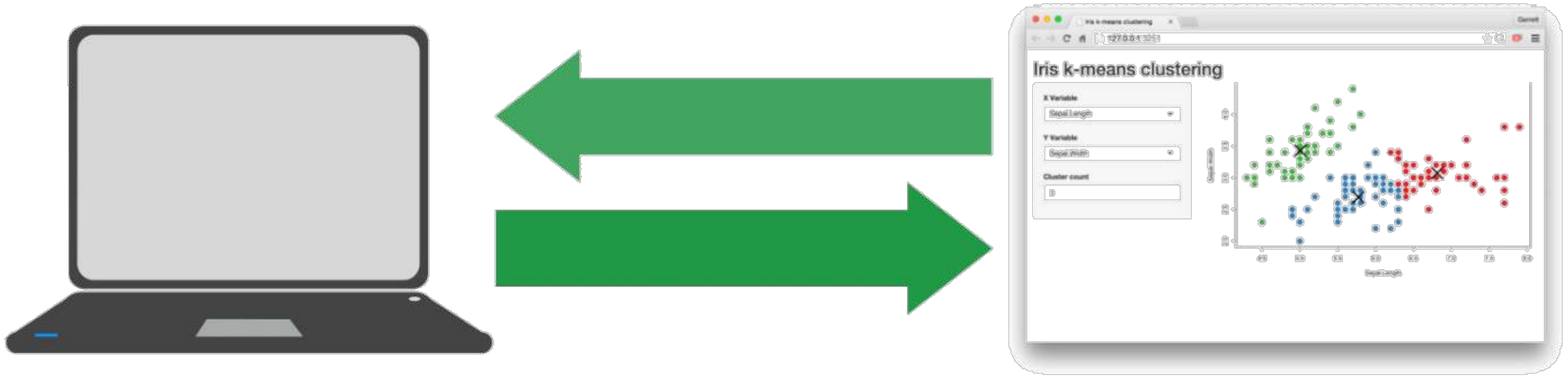
Café de la neuroinformatique

7 juin 2018

What is Shiny ?

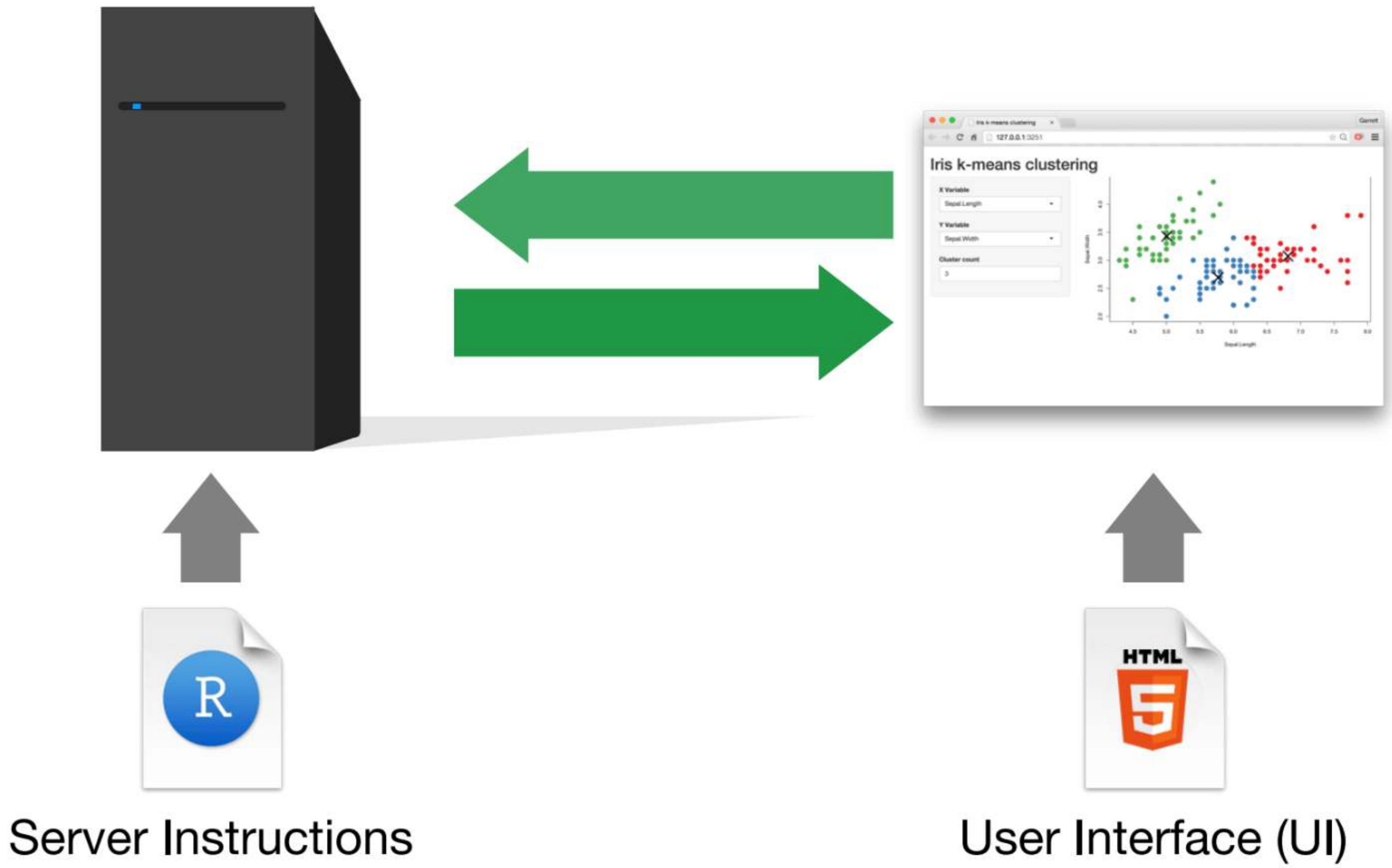
+
Understand the architecture

Every Shiny app is maintained by a computer running R



https://shiny.rstudio.com/tutorial/

Understand the architecture

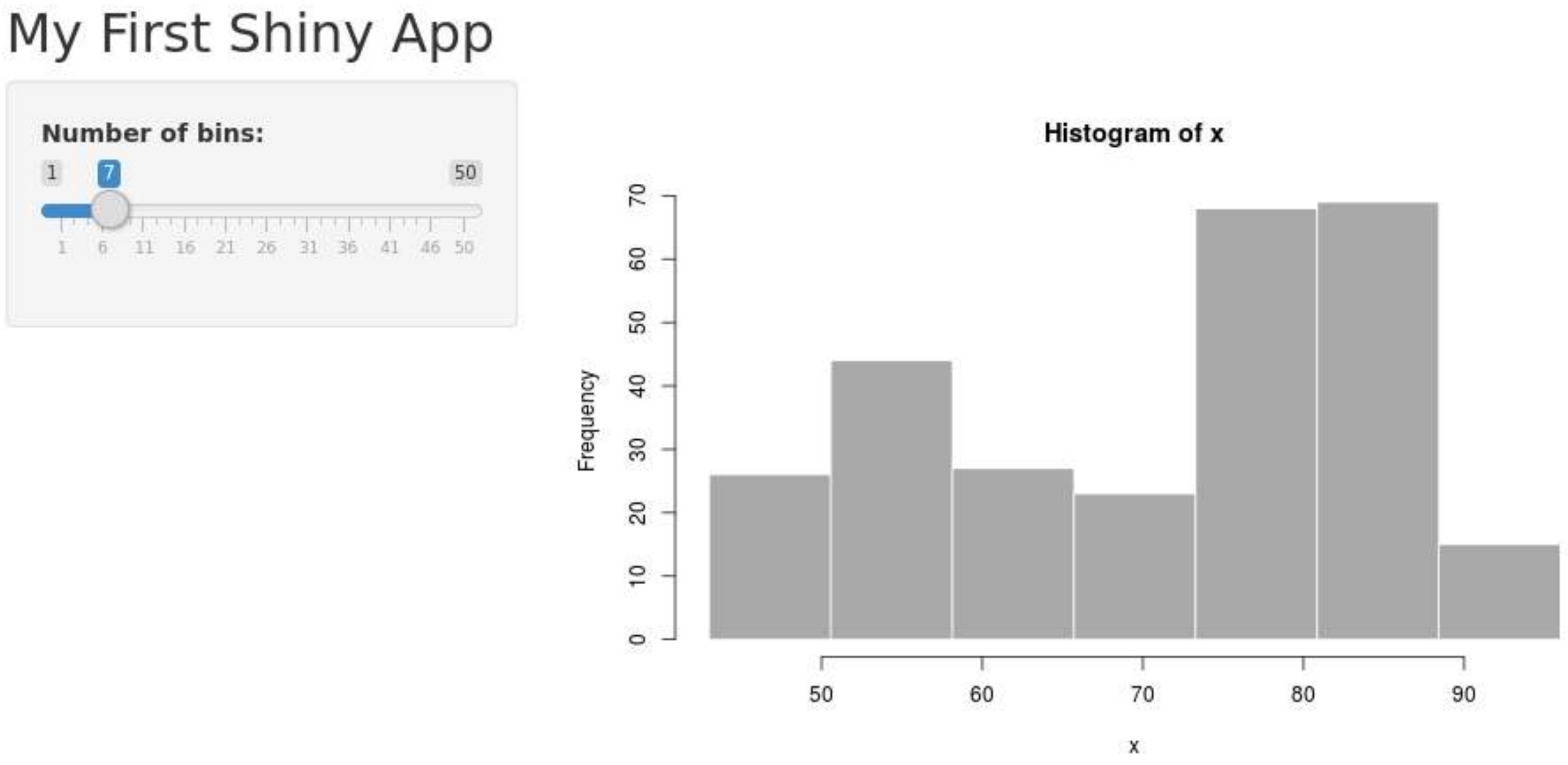


App template

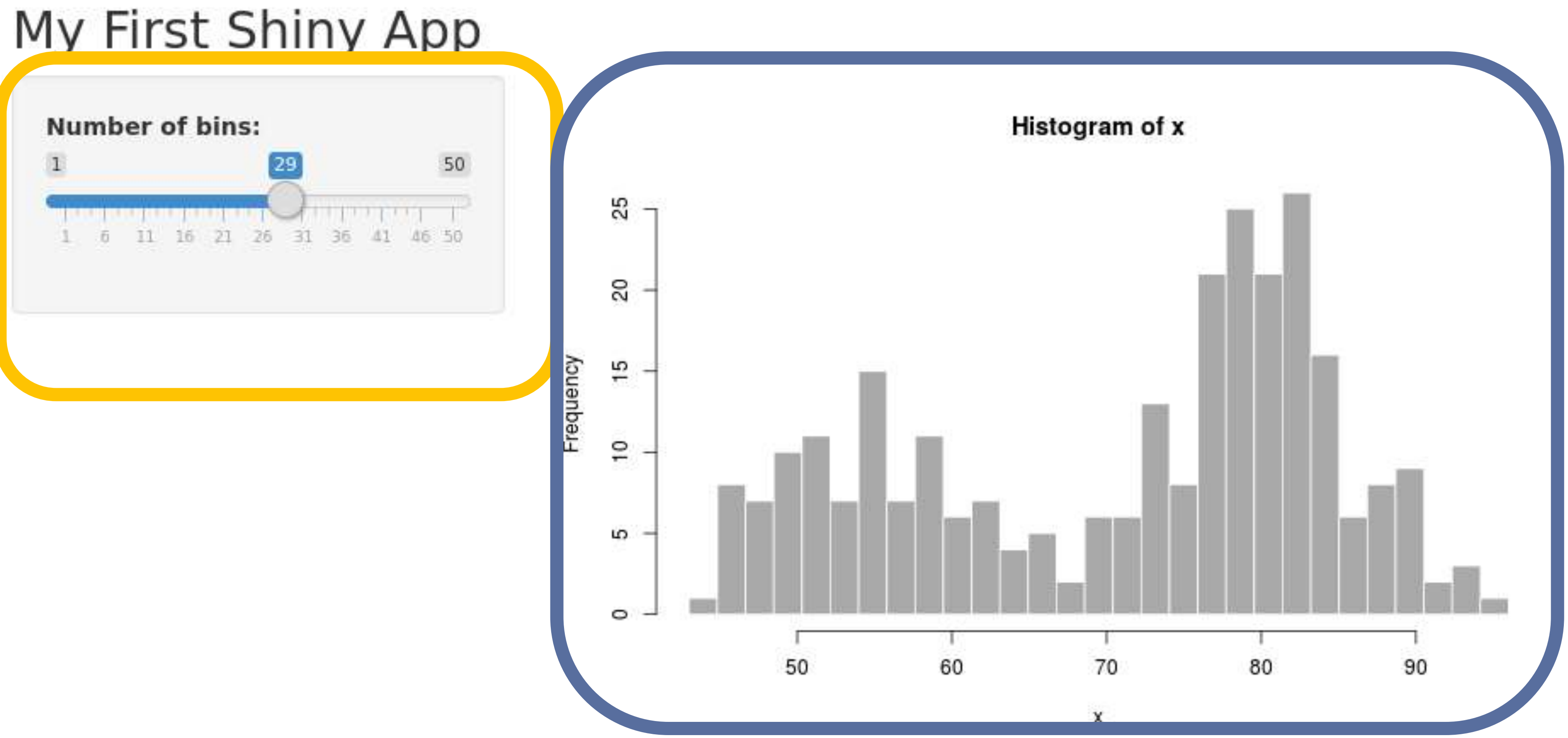
The shortest viable shiny app

```
library(shiny)

ui <- fluidPage()


server <- function(input, output) {}


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

+
Build your app around Inputs and Outputs

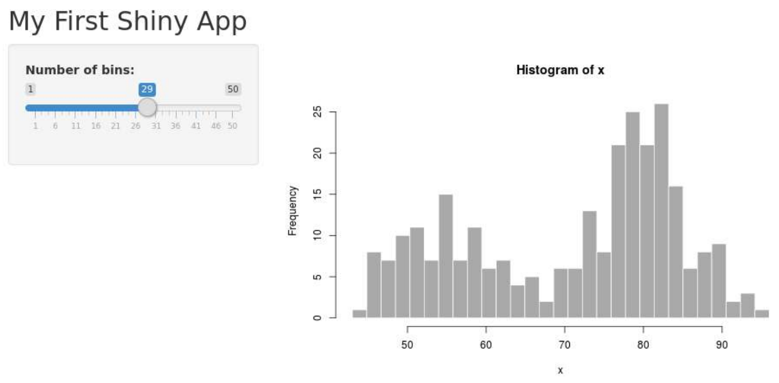


+ Build your app around Inputs and Outputs



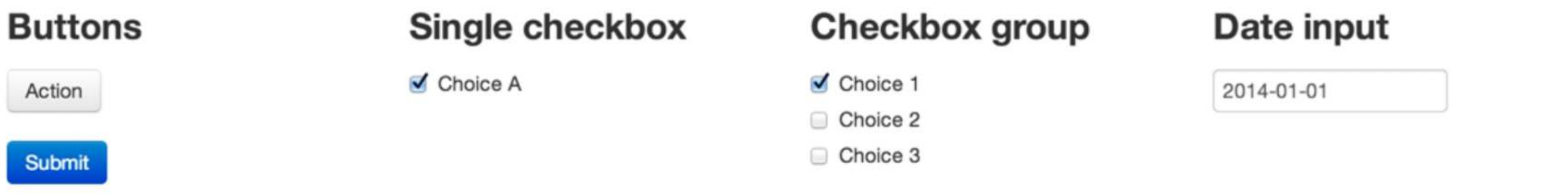
+

Build your app around Inputs and Outputs

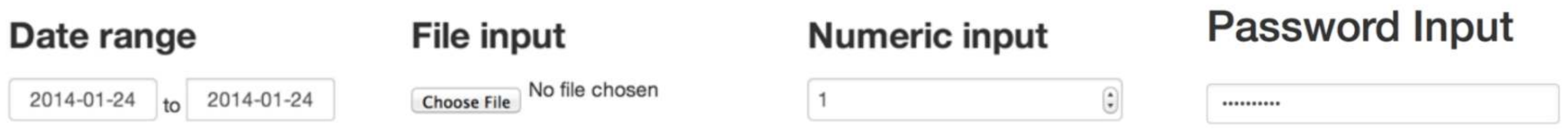


```
library(shiny)
ui <- fluidPage(

  titlePanel("My First Shiny App"),

  sidebarPanel(
    sliderInput("bins”,"Number of bins:”,min=1, max=50, value=30)
  ),

   # Show a plot of the generated distribution
   mainPanel(
     plotOutput("distPlot")
   )
  )
)
```

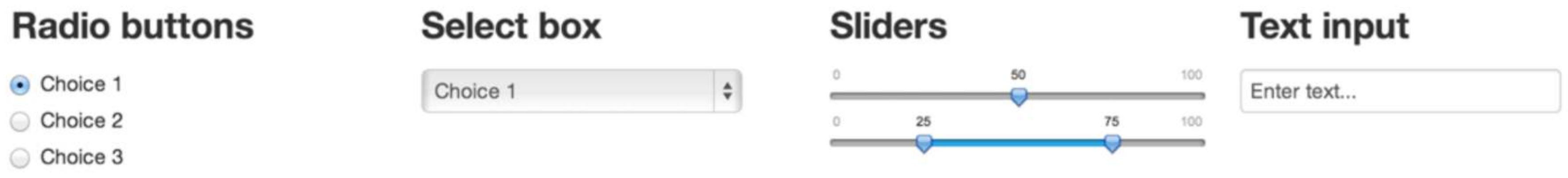
Inputs - Examples



`actionButton()` `submitButton()` `checkboxInput()` `checkboxGroupInput()` `dateInput()`



`dateRangeInput()` `fileInput()` `numericInput()` `passwordInput()`



`radioButtons()` `selectInput()` `sliderInput()` `textInput()`

+
Inputs Syntax

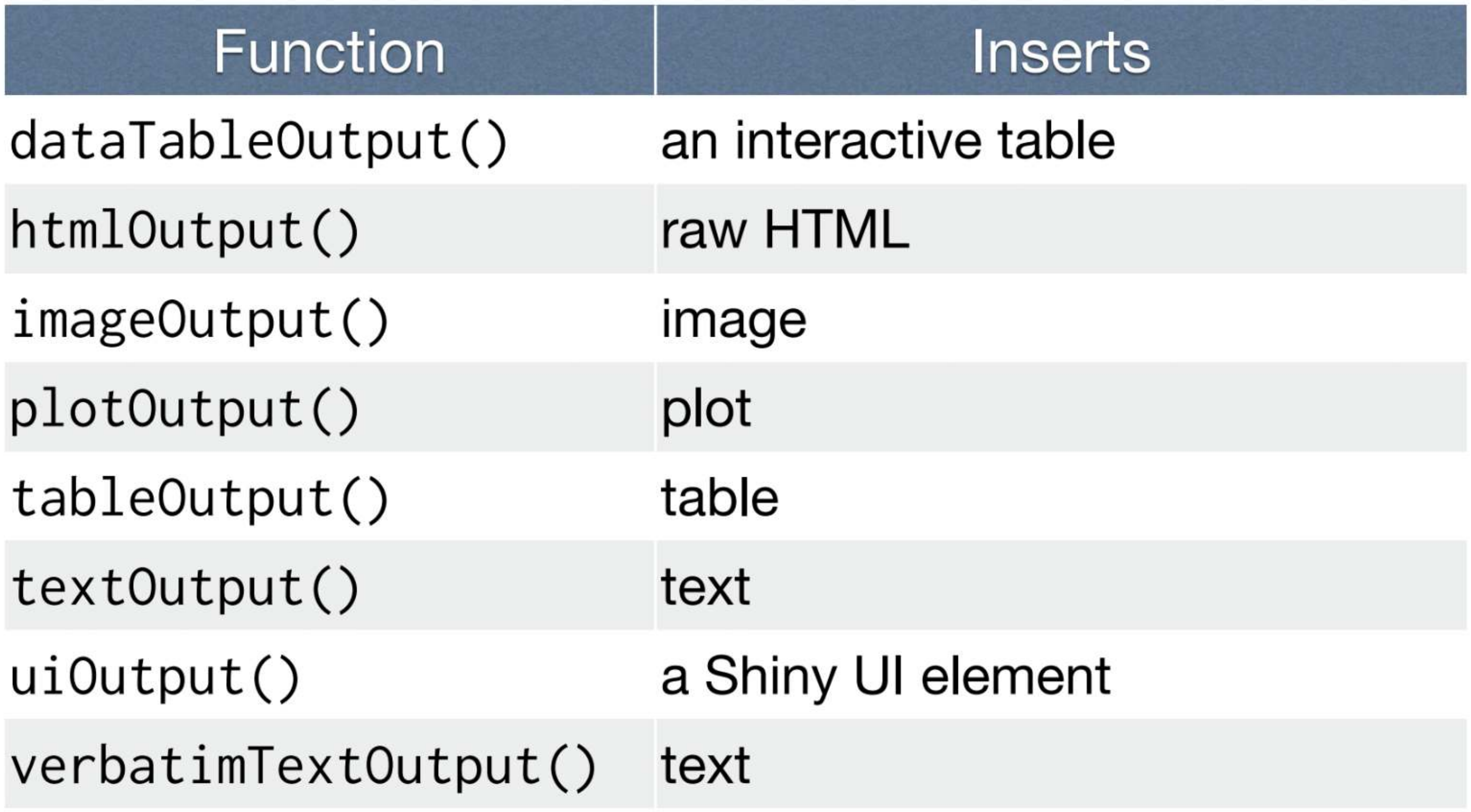
```
sliderInput("bins”,"Number of bins:”, …)
```

Input name
(for internal use)

Label to
display

Input specific
arguments

Outputs function - Examples



Function	Inserts
`dataTableOutput()`	an interactive table
`htmlOutput()`	raw HTML
`imageOutput()`	image
`plotOutput()`	plot
`tableOutput()`	table
`textOutput()`	text
`uiOutput()`	a Shiny UI element
`verbatimTextOutput()`	text

https://shiny.rstudio.com/tutorial/

Outputs Syntax

To display output element, add it in fluidPage() with an output() function

```
plotOutput("distPlot”)
```

The type of output to display

Name to give to the output object

+ Tell Server How to Assemble Inputs into Outputs

```
# Define server logic required to draw a histogram
server <- function(input, output) {

   output$distPlot <- renderPlot({
      # generate bins based on input$bins from ui.R
      x    <- faithful[, 2]
      bins <- seq(min(x), max(x), length.out = input$bins + 1)

      # draw the histogram with the specified number of bins
      hist(x, breaks = bins, col = 'darkgray', border = 'white')
   })
}


# Run the application
shinyApp(ui = ui, server = server)
```

+ Tell Server How to Assemble Inputs into Outputs

- Save objects to display to `output$`

```
output$distPlot  ⟶  plotOutput("distPlot")
```

- Build objects to display with `render()`

```
output$distPlot <- renderPlot({<code block that build object>})
```

- Access input values with `input$`

```
               sliderInput(“num”)
                      ↓
output$distPlot <- renderPlot({hist(rnorm(input$num)
```

+
Live Demo

- Live demo

+

Dash
Create reactive Web in pure Python

Dash

by plotly

http://teaching.lukas-snoek.com

QUESTIONS