Challenge-8

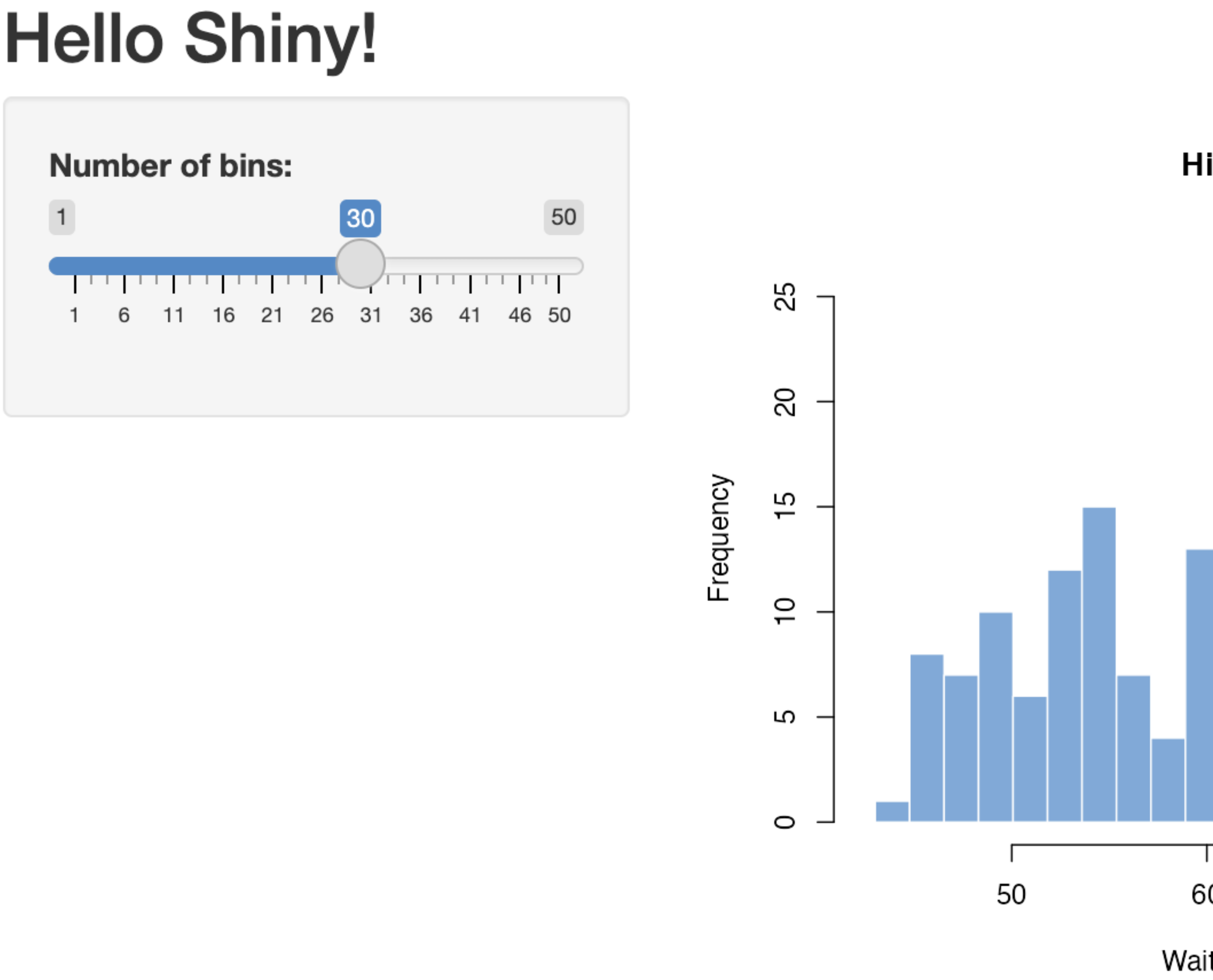
Wang Renhe

09-10-2023

Solution:

```r
# Enter code here
#install.packages("shiny")
library(shiny)
```

```r
knitr::include_graphics("/Users/haley/Desktop/1.png")
```

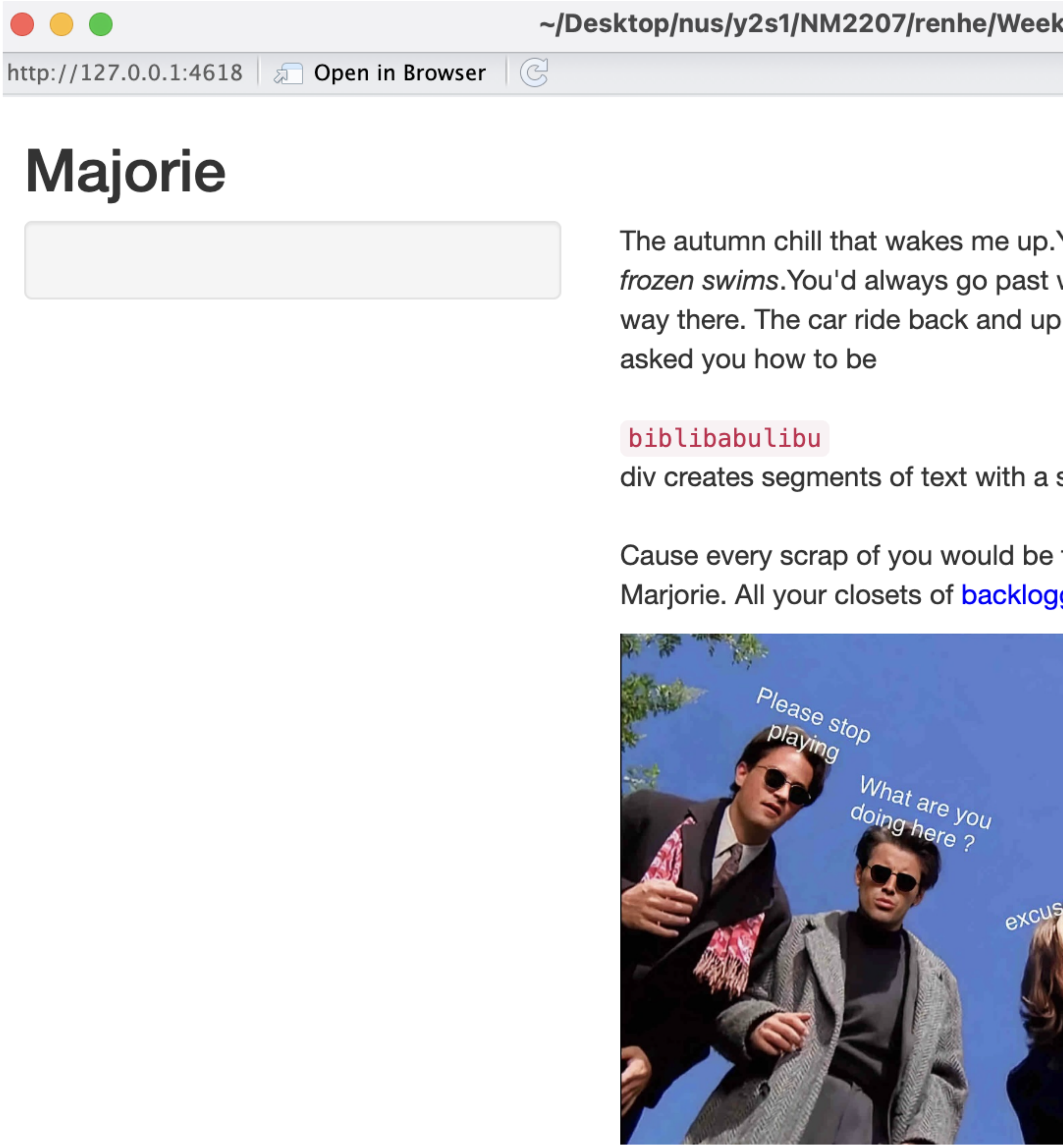


Hello Shiny!

Number of bins:

30

```r
knitr::include_graphics("/Users/haley/Desktop/6.png")
```



~/Desktop/nus/y2s1/NM2207/renhe/Week

http://127.0.0.1:4618 Open in Browser

Majorie

The autumn chill that wakes me up.Y *frozen swims*.You'd always go past w way there. The car ride back and up asked you how to be

`biblibabulibu`

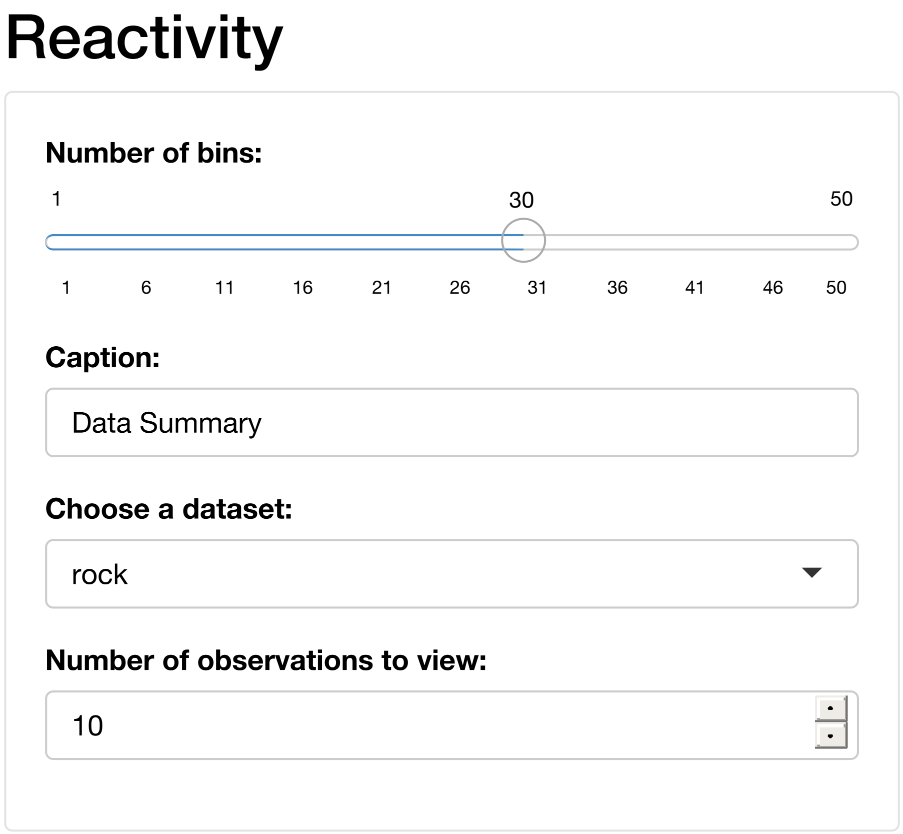
div creates segments of text with a s

Cause every scrap of you would be t Marjorie. All your closets of backlogg

```r
library(shiny)
library(ggplot2)
```

```r
library(shiny)
library(ggplot2)

ui <- fluidPage(
  # App title ----
  titlePanel("Reactivity"),
  # Sidebar layout with input and output definitions ----
  sidebarLayout(
    sidebarPanel(
      sliderInput(inputId = "bins",
                  label = "Number of bins:",
                  min = 1,
                  max = 50,
                  value = 30),
      textInput(inputId = "caption",
              label = "Caption:",
              value = "Data Summary"),
      selectInput(inputId = "dataset",
                label = "Choose a dataset:",
                choices = c("rock", "pressure", "cars")),
      numericInput(inputId = "obs",
                 label = "Number of observations to view:",
                 value = 10),
      ),
    mainPanel(
      plotOutput(outputId = "distPlot"),
      verbatimTextOutput("summary"),
      tableOutput("view")
      )
    )
)



# Define server logic to summarize and view selected dataset ----
server <- function(input, output) {

  datasetInput <- reactive({
    switch(input$dataset,
           "rock" = rock,
           "pressure" = pressure,
           "cars" = cars)
  })

  # Create caption ----
  # The output$caption is computed based on a reactive expression
  # that returns input$caption. When the user changes the
  # "caption" field:
  #
  # 1. This function is automatically called to recompute the output
  # 2. New caption is pushed back to the browser for re-display
  #
  # Note that because the data-oriented reactive expressions
  # below don't depend on input$caption, those expressions are
  # NOT called when input$caption changes
  output$caption <- renderText({
    input$caption
  })

  output$summary <- renderPrint({
    dataset <- datasetInput()
    summary(dataset)
  })

  # Show the first "n" observations ----
  # The output$view depends on both the databaseInput reactive
  # expression and input$obs, so it will be re-executed whenever
  # input$dataset or input$obs is changed
  output$view <- renderTable({
    head(datasetInput(), n = input$obs)
  })


  output$distPlot <- renderPlot({
    x    <- faithful$waiting
    bins <- seq(min(x), max(x), length.out = input$bins + 1)
    hist(x, breaks = bins, col = "#007bc2", border = "white",
         xlab = "Waiting time to next eruption (in mins)",
         main = "Histogram of waiting times")
  })


}

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

```
##
## Listening on http://127.0.0.1:3143
```



Reactivity

Number of bins: 30

Caption: Data Summary

Choose a dataset: rock

Number of observations to view: 10