Week-8

Annette

2023-10-09

```
knitr::opts_chunk$set(echo = TRUE)
```

Code Along

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
#install.packages("shiny")
library(shiny)


ui <- fluidPage(

  tags$head(
    tags$style(HTML("
      body {
        background-color: maroon;
        color: white; /* Set text color to white */
      }
    "))
  ),

  # App title ----
  titlePanel("Annette's Week 8 Challenge"),

  # Sidebar layout with input and output definitions ----
  sidebarLayout(

    # Sidebar panel for inputs ----
```

```
    sidebarPanel(

      # Input: Selector for choosing dataset ----
      selectInput(inputId = "dataset",
                  label = "Choose a dataset:",
                  choices = c("rock", "pressure", "cars")),

      # Input: Numeric entry for number of obs to view ----
      numericInput(inputId = "obs",
                   label = "Number of observations to view:",
                   value = 15)
    ),

    # Main panel for displaying outputs ----
    mainPanel(


      # Output: Verbatim text for data summary ----
      verbatimTextOutput("summary"),

      # Output: HTML table with requested number of observations ----
      tableOutput("view"),

      # Container for the image ----
      div(
        img(src = "rock_image.jpeg", height = 140, width = 300),
        img(src = "car_image.png", height = 140, width = 200)
      )

    )
  )
)

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

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

  # Generate a summary of the dataset ----
  output$summary <- renderPrint({
    dataset <- datasetInput()
    summary(dataset)
  })

  # Show the first "n" observations ----
  output$view <- renderTable({
    head(datasetInput(), n = input$obs)
```

```
  })


}

# Create Shiny app ----
shinyApp(ui = ui, server = server)
```