Intro to Shiny Practical 1

Set-up

Install and load packages

```r
install.packages(c("shiny","ggplot2"))

library(shiny)
library(ggplot2)
```

Exercise 1: Run your first app

```r
library(shiny)

ui <- fluidPage(
  # Allow user to define two numbers
  numericInput(inputId = "add_1", label = "Number:", value = 0),
  numericInput(inputId = "add_2", label = "Number:", value = 0),
  # Display the output
  textOutput(outputId = "sum")
)
server <- function(input, output) {
  output$sum <- renderText({
    input$add_1 + input$add_2
  })
}
shinyApp(ui = ui, server = server)
```

Make sure you can run it both:

- from the console (note: you'll have to save it first)
- using the "Run App" button

Exercise 2: Design a user interface

Open the `prac_1.R` file in the `shiny-practicals` folder.

Identify the opening and closing brackets of the elements of `sidebarLayout` (e.g. `sidebarLayout()`, `sidePanel()`, `mainPanel()`)

Run the app

Make the following changes:

- Change the title of the app to "Demo of a shiny app"

- Add a new selectInput widget to the sidebar, with the options "sex","diet", and "status"

- Add a new textInput widget to the sidebar with the label "Plot title:"

Re-run the app to ensure that the changes have worked

Hint

```r
selectInput(inputId = "input1",
            label = "",
            choices = c("sex","diet","status"))

textInput(inputId = "title",
          label = "")
```

Exercise 3: Create reactive content

Open the `prac_2.R` file in the `shiny-practicals` folder.

Run the app to familiarise yourself with it

Examine the code used to produce the table in the UI/server

- Why is the table not being displayed in the app?
- Fix the code in the UI so that the table is displayed in the `mainPanel`

Examine the code used to produce the plot in the UI/server

- Update the code so that the colour is set by the value of `input$pointcolour`
- Update the code so that the title of the plot is defined by the value of the input from the `inputTextbox` widget

Hint

The colour is currently set to "diet" (`color = "diet"`), so you'll need to set it to use the value of `input$pointcolour`

Similarly, the value of the plot title is currently set to "Plot title" (`title = "Plot title"`), so you'll need to update this to use the value of the inputTextbo widget (i.e. `input$titletext`)

Re-run the app to ensure that the changes have worked