# Building Shiny App exercises part 2

**ADD CONTROL WIDGETS**

In the second part of our series you will see how to add control widgets in your Shiny app. Widget is a web element that your users can interact with. The widgets provided by Shiny are:

**FUNCTIONS**

actionButton: Action Button
checkboxGroupInput: A group of check boxes
checkboxInput: A single check box
dateInput: A calendar for date selection
dateRangeInput: A pair of calendars for selecting a date range
fileInput: A file upload control wizard
helpText: Help text that can be added to an input form
numericInput: A field to enter numbers
radioButtons: A set of radio buttons
selectInput: A box with choices to select from
sliderInput: A slider bar
submitButton: A submit button
textInput: A field to enter text

**ADDING WIDGETS**

You can add widgets to your web page in the same way that you added other types of HTML content in part 1.
To add a widget to your app, place a widget function in sidebarPanel or in mainPanel in your ui.R file.
The first two arguments for each widget are a name for the widget which will be a character string that the user will not see, but you can use it to change the widget's value and a label which will appear with the widget in your app and it should be a character string too.
The rest of the arguments vary from widget to widget,

depending on what the widget needs to be functional. They may be initial values, ranges, and increments.
Follow the examples below to understand the logic behind the widgets' functions and then enhance the app you created in part 1 by practising with the exercise set we prepared for you.
Firstly, we will add all the kinds of the widgets to our app, for educational reasons and later we will decide which of them is practical to keep.
Note that we will just add the buttons in this part. Reactivity will be added to them in a few chapters. Lets begin!

Answers to the exercises are available here.

If you obtained a different (correct) answer than those listed on the solutions page, please feel free to post your answer as a comment on that page.



**Exercise 1**

Open the app you created in part 1 and move the image from sidebarPanel to mainPanel, leave two rows under the title "Main", put the image there and change its dimensions to: height = 150 and width = 200. HINT: Use br.

**BUTTONS**

In the example below we create a UI with a submitButton and an actionButton. Please note that we use the function fluidrow to make sure that all the elements we are going to use will be in the same line as we are going to need this in the next parts:

```
# ui.R
```

```
shinyUI(fluidPage(
titlePanel(“Widgets”),

fluidRow(h3(“Buttons”),
actionButton(“action”, label = “Action”),
br(),
br(),
submitButton(“Submit”))))

# server.R
shinyServer(function(input, output) {
})
```

**Exercise 2**

Leave a row and place an actionButton under the title “Menu” in sidebarPanel, give it the title “Actionbutton”, name = “per” and label = “Perform”. HINT: Use br and h4.

**Exercise 3**

Leave a row from the actionButton you just placed and add a submitButton with title = “Submitbutton” and name = “Submit”. HINT: Use br and h4.

**SINGLE CHECKBOX**
In the example below we create a UI with a single Checkbox:

```
# ui.R
shinyUI(fluidPage(
titlePanel(“Widgets”),

fluidRow(h3(“Single checkbox”),
checkboxInput(“checkbox”, label = “Choice A”, value = TRUE))))

#server.R
shinyServer(function(input, output) {
})
```

**Exercise 4**

Add a checkboxInput in the sidebarPanel under the submitButton, put as title “Single Checkbox”, name it “checbox”, name the label “Choice A” and set the value to “TRUE”. HINT: Use h4.

**Exercise 5**

Change the value to “FALSE” to understand the difference.

**CHECKBOX GROUP**
In the example below we create a UI with a Checkbox Group:

```
#ui.R
shinyUI(fluidPage(
checkboxGroupInput(“checkGroup”,
label = h3(“Checkbox group”),
choices = list(“Choice 1” = 1,
“Choice 2” = 2, “Choice 3” = 3),
selected = 2)
))

#server.R
shinyServer(function(input, output) {
})
```

**Exercise 6**

Place a checkboxGroupInput under the checkboxInput, give it title “Checkbox Group”, name it “checkGroup”, name the label “Checkbox Group” and give it 3 choices. HINT: Use h4

**Exercise 7**

Make the second of the choicesthe default one.

**DATE INPUT**
In the example below we create a UI with a Date Input:

```
#ui.R
shinyUI(fluidPage(
dateInput(“date”,
label = h3(“Date input”),
```

```
value = “2016-12-07”)
))

#server.R
shinyServer(function(input, output) {
})
```

**Exercise 8**

Under the checkboxGroupInput add a dateInput with name = “date”, label = “Date Input” and value = “2016-12-01”.

**DATE RANGE**
In the example below we create a UI with a Date Range Input:

```
#ui.R
shinyUI(fluidPage(
dateRangeInput(“dates”, label = h3(“Date range”))
))

#server.R
shinyServer(function(input, output) {
})
```

**Exercise 9**

Under the dateInput place a dateRangeInput with name = “dates” and label = “Date Range”. HINT: Use h4.

**FILE INPUT**
In the example below we create a UI with a File Input.

```
#ui.R
shinyUI(fluidPage(
fileInput(“file”, label = h3(“File input”))
))

#server.R
shinyServer(function(input, output) {
})
```

**Exercise 10**

Under the dateRangeInput place a fileInput. Name it “file” and give it the label “File Input”.