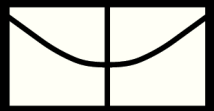


Universidade de Brasília

SHINY

INTERATIVIDADE COM R

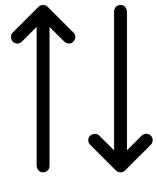
Davi, Eduardo, Gabriela, Jadson, Tailine

SUMÁRIO


Shiny

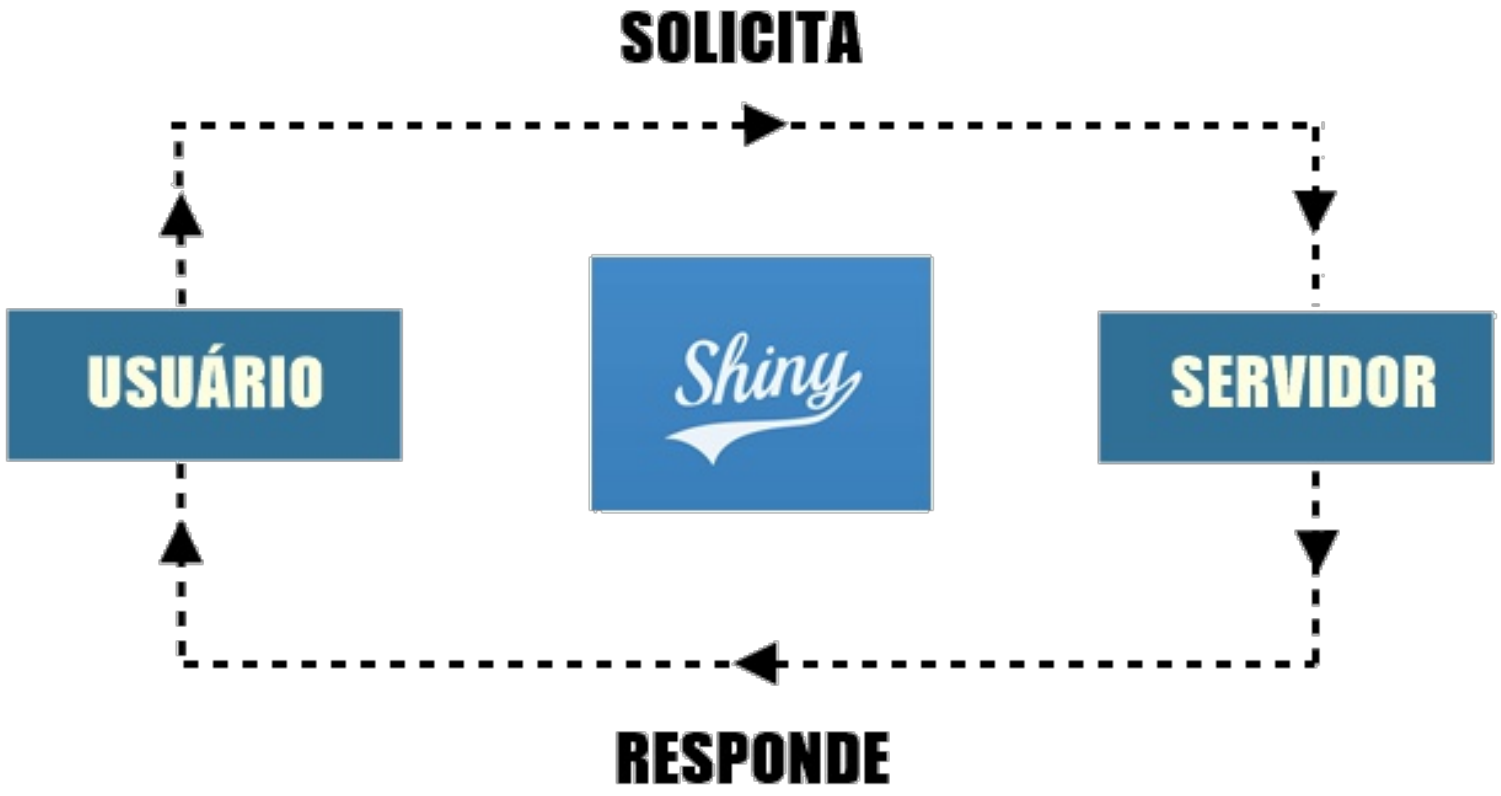
INTERATIVIDADE

Instruções do Servidor (R)



User Interface (UI)

REATIVIDADE



RESUMINDO

Pacote do R

Cria de um servidor que envia páginas web, **recebe** informações do usuário e **processa** os dados.

Permite estruturar a interface do usuário **e** o processamento de dados.

Vantagens para o programador e para o usuário.

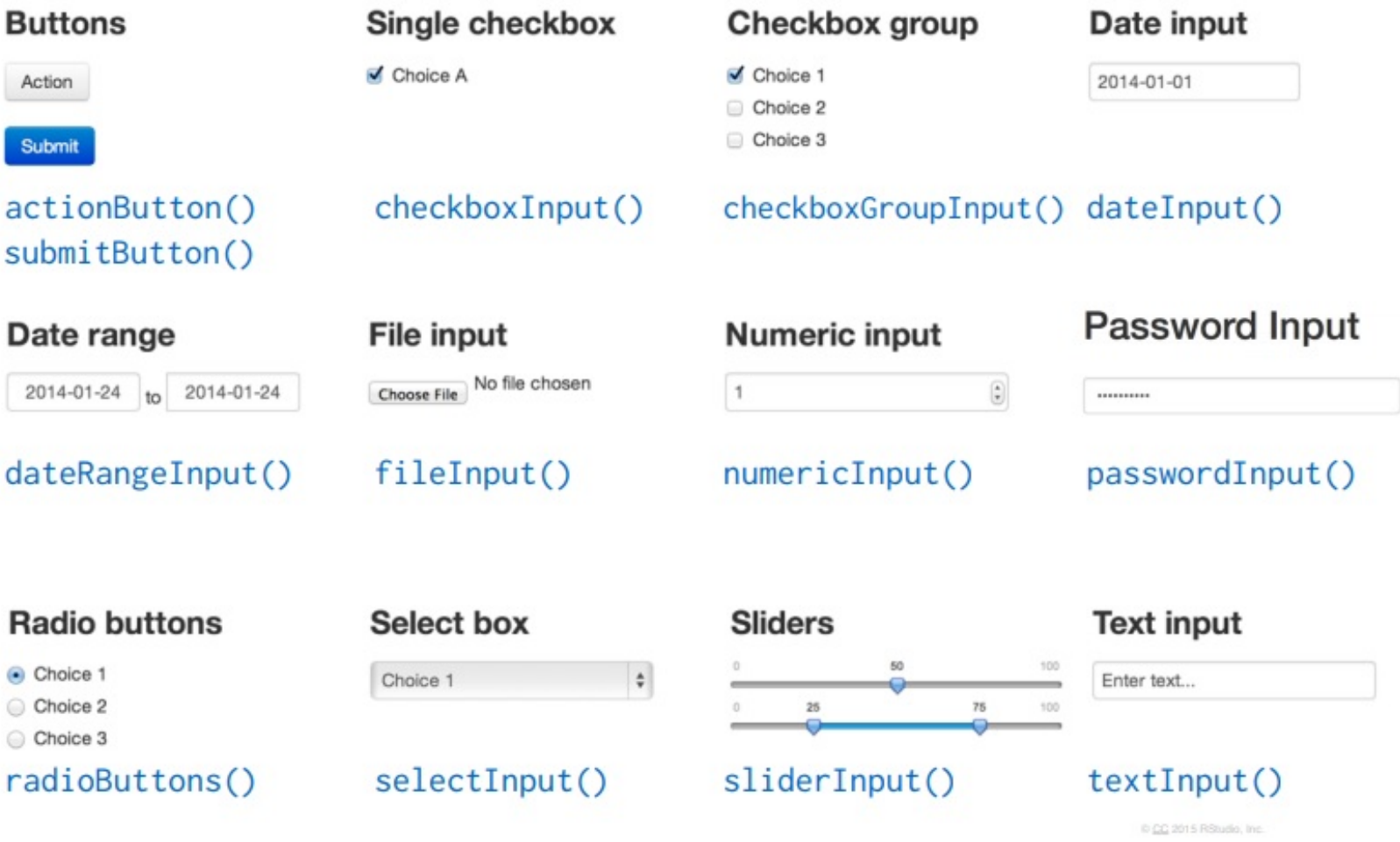
ESTRUTURA

```
library(shiny)

ui <- fluidPage()

server <- function(input, output) {}

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

USER INTERFACE (UI)

Função	Finalidade
library(shiny)	Carregar o pacote Shiny.
ui <- fluidPage()	Criar uma interface com o usuário.
titlePanel()	Criar um painel contendo um título do aplicativo.
sidebarLayout()	Criar um layout com uma barra lateral e área principal.
sidebarPanel()	Criar um painel com barra lateral.
mainPanel()	Criar um painel principal contendo elementos de saída.

WIDGETS



Fonte: Shiny from RStudio

CRIANDO FUNÇÕES DE ENTRADA

```
library(shiny)

ui <- fluidPage(

  sliderInput(inputId = ”num”,

              label = NULL,

              value = 25, min = 1, max = 100) )

server <- function(input, output) {}

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

CRIANDO FUNÇÕES DE ENTRADA

```
library(shiny)

ui <- fluidPage(
  sliderInput(inputId = "num",
              label = NULL,
              value = 25, min = 1, max = 100) )

server <- function(input, output) {}

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

PRÓXIMO PASSO

Para que seja possível visualizar o input, é necessário escolher como será o output. Para esse exemplo, queremos que o output gere um gráfico. Mas que função precisamos usar agora?

OUTPUTS

Função	Finalidade
dataTableOutput()	Tabela Interativa
htmlOutput()	HTML puro
imageOutput()	Imagem
plotOutput()	Gráfico
tableOutput()	Tabela
textOutput()	Texto
uiOutput()	Elemento do Shiny UI
verbatimTextOutput()	Texto

DEFININDO O TIPO DE OUTPUT

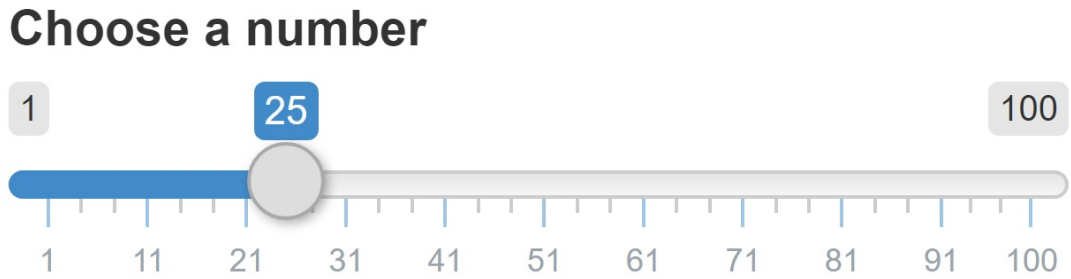
```
library(shiny)
ui <- fluidPage(
  sliderInput(inputId = ”num”,
              label = NULL,
              value = 25, min = 1, max = 100),
  plotOutput(”hist”))
server <- function(input, output) {}
shinyApp(ui = ui, server = server)
```

DEFININDO O TIPO DE OUTPUT

```
library(shiny)

ui <- fluidPage(

  sliderInput(inputId = "num",

              label = NULL,

              value = 25, min = 1, max = 100),

  plotOutput("hist"))

server <- function(input, output) {}

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

RESULTADO

Agora foi gerado um botão de slide onde o usuário fará a escolha de um número entre 1 e 100.



PRÓXIMO PASSO

A próxima etapa é configurar o output.

Dentro do UI, apenas demos alguns nomes.

Agora precisamos definir o que realmente vai acontecer.

SERVER

Função	Finalidade
library(shiny)	Carregar o pacote Shiny.
shinyServer()	Definir a lógica do servidor do aplicativo Shiny.
function(input,output)	Funções render()

RENDER ()

Output (UI)	Render (Server)
dataTableOutput()	renderDataTable
imageOutput()	renderImage
plotOutput()	renderPlot
tableOutput()	renderTable
textOutput()	renderText
verbatimTextOutput()	renderPrint
uiOutput()	renderUI
htmlOutput()	renderUI

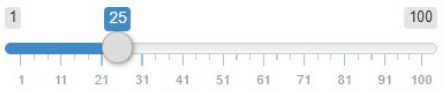
CONFIGURANDO O OUTPUT

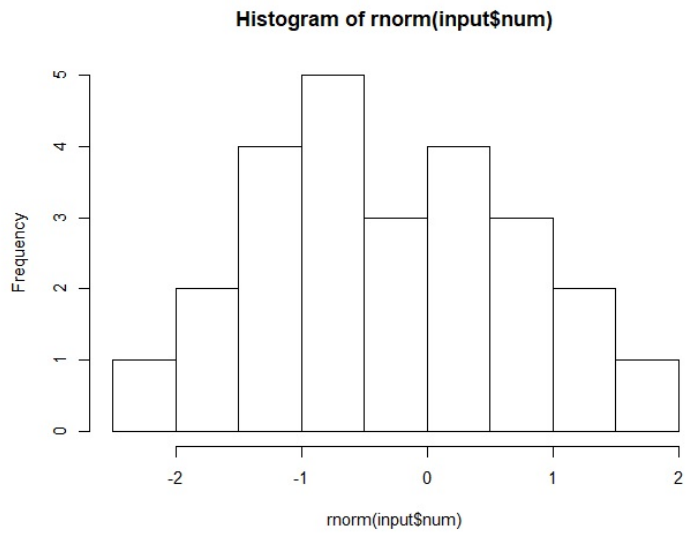
```
library(shiny)
ui <- fluidPage(
  sliderInput(inputId = ”num”,
              label = NULL,
              value = 25, min = 1, max = 100),
  plotOutput(”hist”))
server <- function(input, output) {
  output$hist <- renderPlot({
    hist(rnorm(input$num))})}
shinyApp(ui = ui, server = server)
```

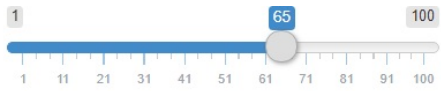
CONFIGURANDO O OUTPUT

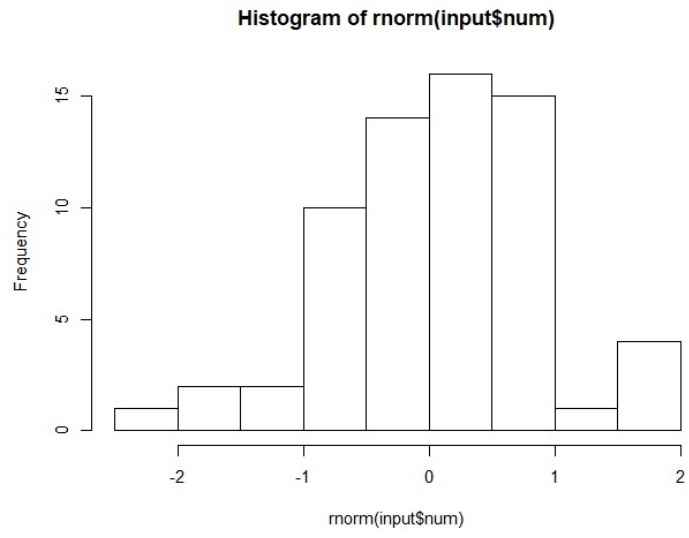
```
library(shiny)

ui <- fluidPage(

  sliderInput(inputId = "num",

              label = NULL,

              value = 25, min = 1, max = 100),

  plotOutput("hist"))

server <- function(input, output) {

  output$hist <- renderPlot({

    hist(rnorm(input$num))})}

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

RESULTADO


1
25
100
1 11 21 31 41 51 61 71 81 91 100


Histogram of rnorm(input$num)
Frequency
0 1 2 3 4 5
-2 -1 0 1 2
rnorm(input$num)


1
65
100
1 11 21 31 41 51 61 71 81 91 100


Histogram of rnorm(input$num)
Frequency
0 5 10 15
-2 -1 0 1 2
rnorm(input$num)

APLICAÇÕES

REFERÊNCIAS

1. RSTUDIO INC. **Shiny from RStudio**. Disponível em: https://shiny.rstudio.com/tutorial/. Acesso em: setembro de 2019.
2. PUC MINAS. **Desenvolvimento de Aplicativos Web Com R e Shiny:** inovações no ensino de Estatística. Belo Horizonte,v. 6, n. 2, p. 55-71, maio 2018
3. **Curso-R**. Disponível em: http://material.curso-r.com/shiny/. Acesso em: setembro de 2019.