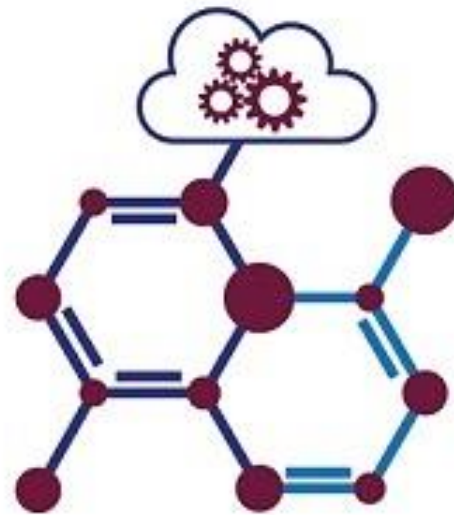




TECNOLÓGICO  
NACIONAL DE MÉXICO®



CIENCIA DE  
LOS DATOS

Diplomado en:

# CIENCIA DE LOS DATOS E INTERNET DE LAS COSAS

## VIRTUAL





MTI. Jose Gabriel Rodríguez Rivas, LI



MTI. Marzo Antonio Rodríguez Zúñiga, ISC



MAI. Rubén Pizarro Gurrola, LI

# Objetivo

- Conocer y aplicar controles básicos de R Shiny

# Temas

Temas	Estimado
<ol style="list-style-type: none"><li>1. Controles de entrada</li><li>2. Entradas de texto libre</li><li>3. Entradas numéricas</li><li>4. Entradas de selección y radio botón</li><li>5. Entradas de casillas verificaciones</li><li>6. Controles de salida (output)</li><li>7. Creación de aplicaciones con asistente</li><li>8. Publicación de aplicaciones shiny</li><li>9. Documentos interactivos markdown con shiny</li></ol>	2

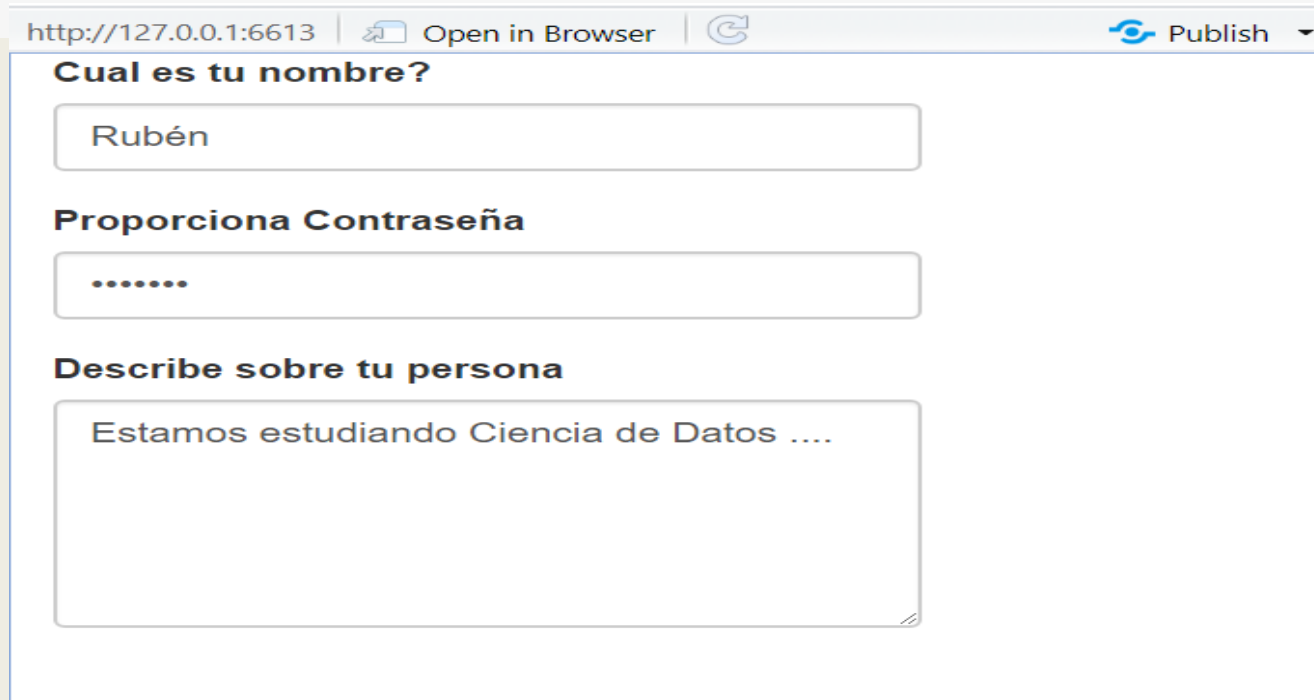


# 1. Controles de entrada

- Son controles que se especifican en la función `ui()` del documento R Shiny
- Cada control tiene restricciones de tener un identificador además de ser único.
- El identificador se establece por el argumento `inputId` = “*nombre*”, el nombre deben ser caracteres, números o guines bajos no caracteres especiales. [acentos, ñs, espacios...]
- Algunos controles tienen el atributo `label` = “*valor*” que es como la etiqueta asociada al control de entrada.
- Los controles se acceden en la función `server()` con `input$nombre`
- Todos los `input` y `output` van dentro de `ui(....)` sepradados por coma.

## 2. Entradas de texto libre

```
ui <- fluidPage(  
  textInput("nombre", "Cual es tu nombre? "),  
  passwordInput("contras", "Proporciona Contraseña"),  
  textAreaInput("acerca", "Describe acerca de tu persona", rows = 5)  
)
```



http://127.0.0.1:6613 | Open in Browser | Publish

**Cual es tu nombre?**

Rubén

**Proporciona Contraseña**

.....

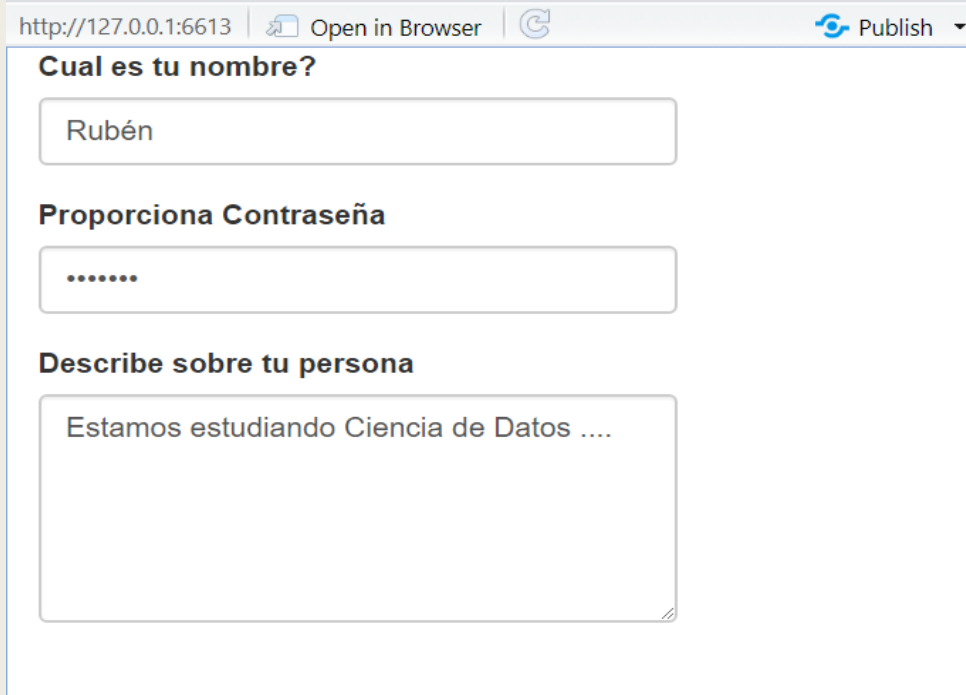
**Describe sobre tu persona**

Estamos estudiando Ciencia de Datos ....

# Desafío 8.

*Crear el formulario con los controles de texto libre y mostrar lo que se captura en la caja de texto del nombre y del area de texto en la función `server()` como una salida y como una tabla.*

*Llamar al archivo **08.desafio.r***



http://127.0.0.1:6613 | Open in Browser | Publish

**Cual es tu nombre?**

Rubén

**Proporciona Contraseña**

.....

**Describe sobre tu persona**

Estamos estudiando Ciencia de Datos ....

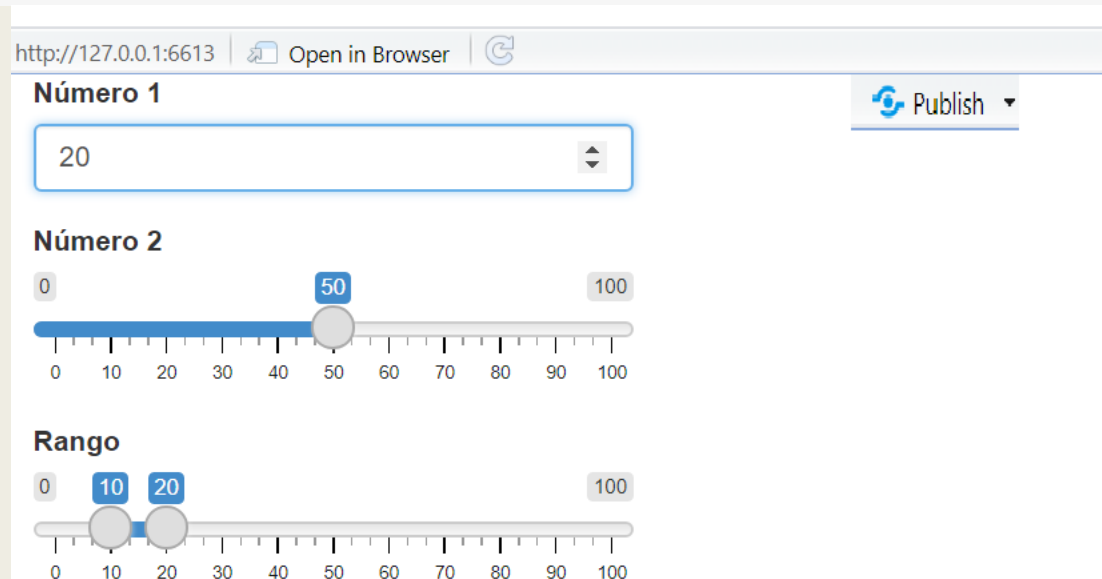


Mostrar los valores de nombre y acerca de los controles de captura en formato texto y tabla respectivamente



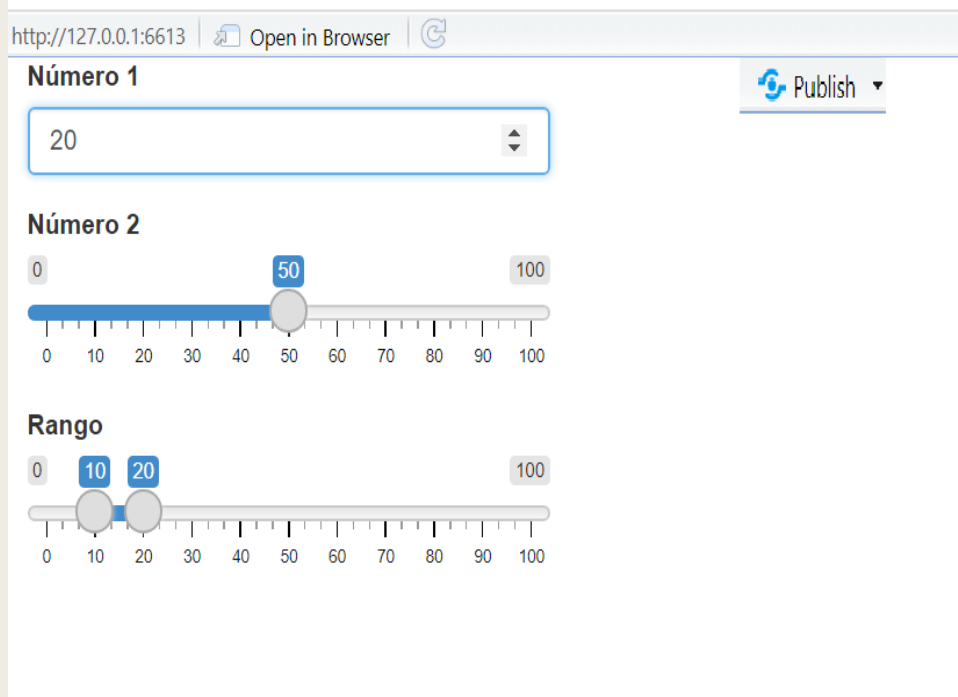
### 3. Entradas numéricas

```
ui <- fluidPage(  
  numericInput("num1", "Número 1", value = 50, min = 1, max = 100),  
  sliderInput("num2", "Número 2", value = 50, min = 1, max = 100),  
  sliderInput("rng", "Rango", value = c(10, 20), min = 1, max = 100)  
)
```



# Desafío 09

Generar una muestra de números entre el **rango** proporcionado por rango y la cantidad de números establecidos en el valor de cualquiera de el control **num1** y utilizar el control **num2** como semilla **set.seed(num2)**  
Mostrar los datos de la muestra en un **plot()** diagrama de dispersión  
Llamar al archivo **09.desafio.r**



http://127.0.0.1:6613 Open in Browser

Número 1

20

Número 2

0 50 100

Rango

0 10 20 100

Publish



Mostrar los números en modo consola o textual y un **plot(datos(), type, 'b')** de la dispersión con puntos y líneas al mismo tiempo el histograma con **hist(datos())**

# Desafío 10

Semejante al desafio09. Generar una distribución normal que representa valores de edades de personas.

La cantidad de números igual a  $n$

La media igual al valor de la entrada numérica slider llamada **media**

La desviación igual al valor de la entrada numérica slider llamada **desv**,

Utilizar el control slider llamado semilla como semilla **set.seed(num2)**

Mostrar los datos de la muestra en un **ggplot()** con la curva de densidad

Lamar al archivo **10.desafio.r**

http://127.0.0.1:6613 | Open in Browser |

Cantidad de números

Publish

La media de la distribución

0 50 100

0 10 20 30 40 50 60 70 80 90 100

Desviación estándar de la distribución

5 20 30

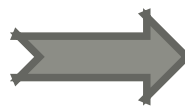
5 10 15 20 25 30

Semilla para valores aleatorios de la distribución normal

1 50 100

1 11 21 31 41 51 61 71 81 91 100

61 33 51 60 15 44 57 38 70 21 56 61 40 54 41 43 47 35 27 44 43 38 18 84 61 103 57 43 61 51 54 33 27 62 28 50 58 87 58 58 52 42 52 -8 52 60 69 17 20 40



Visualizar los números generados  
la dispersión y el histograma con  
**plot()**

## 4. Entradas de selección y radio botón

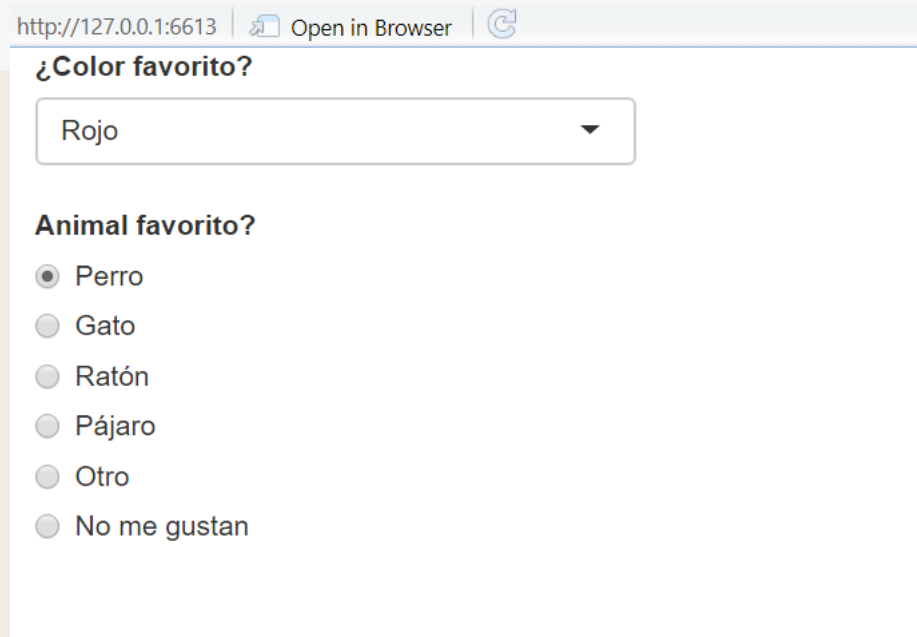
```
library(shiny)
```

```
# Los datos
```

```
animals <- c("Perro", "Gato", "Ratón", "Pájaro", "Otro", "No me gustan")
```

```
colores <- c("Rojo", "Azul", "Amarillo", "Verde", "Otro", "Otro")
```

```
ui <- fluidPage(  
  selectInput(inputId = "colores", label = "¿Color favorito?", choices = colores),  
  radioButtons(inputId = "animales", label = "Animal favorito? ", choices = animals)  
)
```



The screenshot shows a web browser window with the URL `http://127.0.0.1:6613`. The page contains two interactive elements:

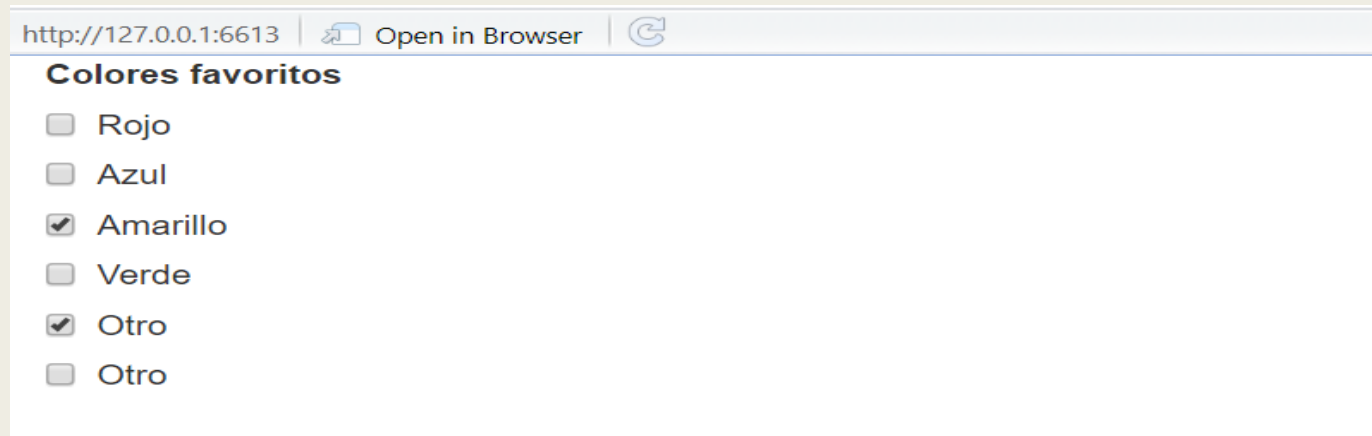
- A dropdown menu titled "¿Color favorito?" with "Rojo" selected.
- A group of radio buttons titled "Animal favorito?". The options are: Perro (selected), Gato, Ratón, Pájaro, Otro, and No me gustan.

# 5. Entradas de Casillas de verificación

# Los datos

```
colores <- c("Rojo", "Azul", "Amarillo", "Verde", "Otro", "Otro")
```

```
ui <- fluidPage(  
  checkboxGroupInput(inputId = "colores", label = "Colores favoritos",  
    choices = colores)  
)
```



http://127.0.0.1:6613 | Open in Browser

**Colores favoritos**

- ☐ Rojo
- ☐ Azul
- ☒ Amarillo
- ☐ Verde
- ☒ Otro
- ☐ Otro

## 6. Controles de Salidas (output text)

- Hay tres tipos principales de resultados, que se corresponden con las tres cosas que normalmente incluye en un informe: **texto**, **tablas** y **gráficos**.
- Con respecto a salidas texto:
  - *texto normal con `textOutput ()` con salida `renderText()` texto*
  - *código fijo y salida de consola con `verbatimTextOutput ()` con salida `renderPrint()` como si fuera consola.*



## 6. Controles de Salidas (output table)

- Con respecto a salidas tipo tabla:
  - *tableOutput ()* con su salida *renderTable ()* representan una tabla estática de datos, mostrando todos los datos a la vez
  - *dataTableOutput ()* con su salida *renderDataTable ()* representan una tabla dinámico con controles para cambiar comportamientos.

nombres	edades	estados
Jorge	35	Sinaloa
LUIS	37	Coahuila
Carlos	64	Sonora
Paty	43	Chihuahua
Carmen	28	Jalisco
Rubén	51	Durango

Show 25 ▼ entries

Search:

nombres	edades	estados
Rubén	51	Durango
JESÚS	33	Durango
FELIPE	33	Durango
MARÍA	33	Durango
MANUEL	30	Durango
GABRIELA	39	Durango
ROSA	55	Durango
GUILLERMO	40	Durango
JAVIER	38	Durango
MARTHA	46	Durango
MARCO ANTONIO	39	Durango

nombres edades DURANGO

Showing 1 to 11 of 11 entries (filtered from 100 total entries)

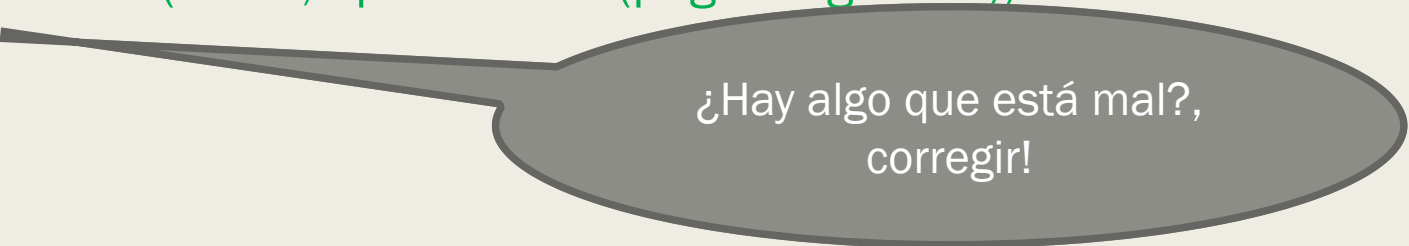
Previous 1 Next

## 6. Controles de Salidas (output table)

- 11.desafio.r
- Crear dos salidas tipo tabla usando *tableOutput()* y *dataTableOutput()* mostrando los datos de la fuente: <https://raw.githubusercontent.com/rpizarrog/diplomado-cd-iot-2021/main/funciones/crear%20datos.r>

```
library(shiny)
source("https://raw.githubusercontent.com/rpizarrog/diplomado-cd-iot-2021/main/funciones/crear%20datos.r" )

ui <- fluidPage(
  tableOutput("fijo"),
  dataTableOutput("dinámico")
)
server <- function(input, output, session) {
  output$static <- renderTable(head(datos))
  output$dynamic <- renderDataTable(datos, options = list(pageLength = 5))
}
shinyApp(ui, server)
```



¿Hay algo que está mal?,  
corregir!

## 6. Controles de Salidas (output plot)

- Pueden desplegar cualquier gráficos con la función base de `plot()` o gráficos con `ggplot()`
- El control de salida `plotOutput()` va a sociadao a la función `renderPlot()`

### 12.desafio.r

Crear una gráficas de barras de la frecuencia de estados o con `plot()` o con `ggplot()` de los datos:

<https://raw.githubusercontent.com/rpizarrog/diplomado-cd-iot-2021/main/funciones/crear%20datos.r>

# 7. Creando aplicaciones Shiny con asistente

- File/New/Shiny Web App
- Crear una aplicación shiny por asistente en carpeta **13.desafio**

# 8. Publicación de aplicaciones

- Publicar una aplicación en la nube [Shinyapps.io](https://shinyapps.io)
- 14.desafio
- Pendiente ...

## 9. Documentos interactivos markdown con shiny

- Crear un documento markdown e incrustar funciones shiny
- 15.desafio.rmd
- Pendiente



# Controles de posición `fluidRow(column())`

- Funciones para crear diseños de página.
- Un diseño de posición e elementos `fluid()` consta de filas que a su vez incluyen columnas.
- Las filas existen con el propósito de asegurarse de que sus elementos aparezcan en la misma línea (si el navegador tiene el ancho adecuado).
- Las columnas existen con el propósito de definir cuánto espacio horizontal dentro de una cuadrícula de 12 unidades de ancho deben ocupar sus elementos.

# Controles de posición fluidRow(column())

ui <-

fluidPage(

fluidRow(

column(width = 3,

"columna 1 de width 4",

numericInput("n", "Cantidad de números", value = 50, min = 10, max = 500, step = 20),

sliderInput("media", "La media de la distribución", value = 50, min = 0, max = 100, step = 10),

sliderInput("desv", "Desviación estándar de la distribución", value = 20, min = 5, max = 30, step = 5),

sliderInput("semilla", "Semilla para valores aleatorios de la distribución normal", value = 50, min = 1, max = 100, step = 10),

),

column(width = 7,

"Columna 2 de width 3", textOutput("numeros"),

plotOutput("grafica")

)

)

)



Se reutiliza 10.desafio.r

# Controles de posición fluidRow(column())

```
server <- function(input, output, session) {
```

```
  datos <- reactive({
```

```
    set.seed(input$semilla)
```

```
    round(rnorm(n = input$n, mean = input$media, sd = input$desv),0)
```

```
  })
```

```
  output$numeros <- renderText({
```

```
    datos() })
```

```
  output$grafica <- renderPlot({
```

```
    par(mfrow=c(1,3))
```

```
    plot(datos(), type='b')
```

```
    hist(datos())
```

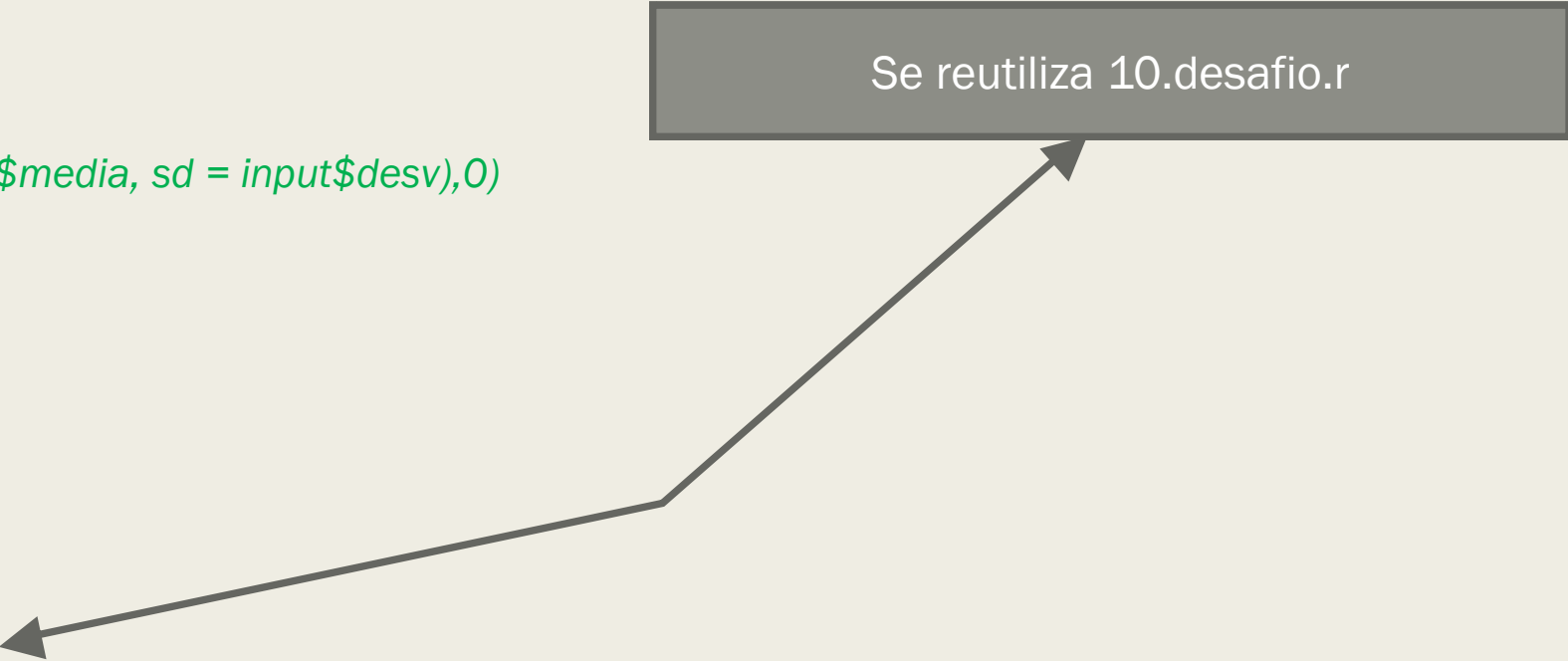
```
    hist(datos(), freq = FALSE)
```

```
    lines(density(datos(), col = "red"))
```

```
    par(mfrow=c(1,1))
```

```
  }) }
```

Se reutiliza 10.desafio.r



# Bibliografía

- Wickhan, H. (2020). Mastering Shiny. (C. C.-N.-N. License., Ed.) USA: O'Reilly Media. Obtenido [de https://mastering-shiny.org/basic-ui.html](https://mastering-shiny.org/basic-ui.html)