

# UNA INTRODUCCIÓN A LAS SHINY APPS

**¿Qué son? ¿Para qué sirven? ¿Cómo hago una?**

Por Rui Su y Carla Solis Uehara

# AGENDA PARA HOY

- Presentación de las panelistas.
- Preámbulo: saber R es importante.
- Qué son las shiny apps. Casos de uso.
- Para qué sirven: Qué problemas resuelven
- Nuestro ejemplo personal.
- Cómo hago una: Tutorial para crear una shiny app.



# ¿QUIÉNES SOMOS?



Resolvemos problemas de  
analítica con código

# LA IMPORTANCIA DE R

- saber R es un prerequisite importante para aprender a hacer shiny apps.



# TIDYVERSE Y LA CIENCIA DE DATOS

Consejo:



Tidyverse en español  
<https://es.r4ds.hadley.nz>



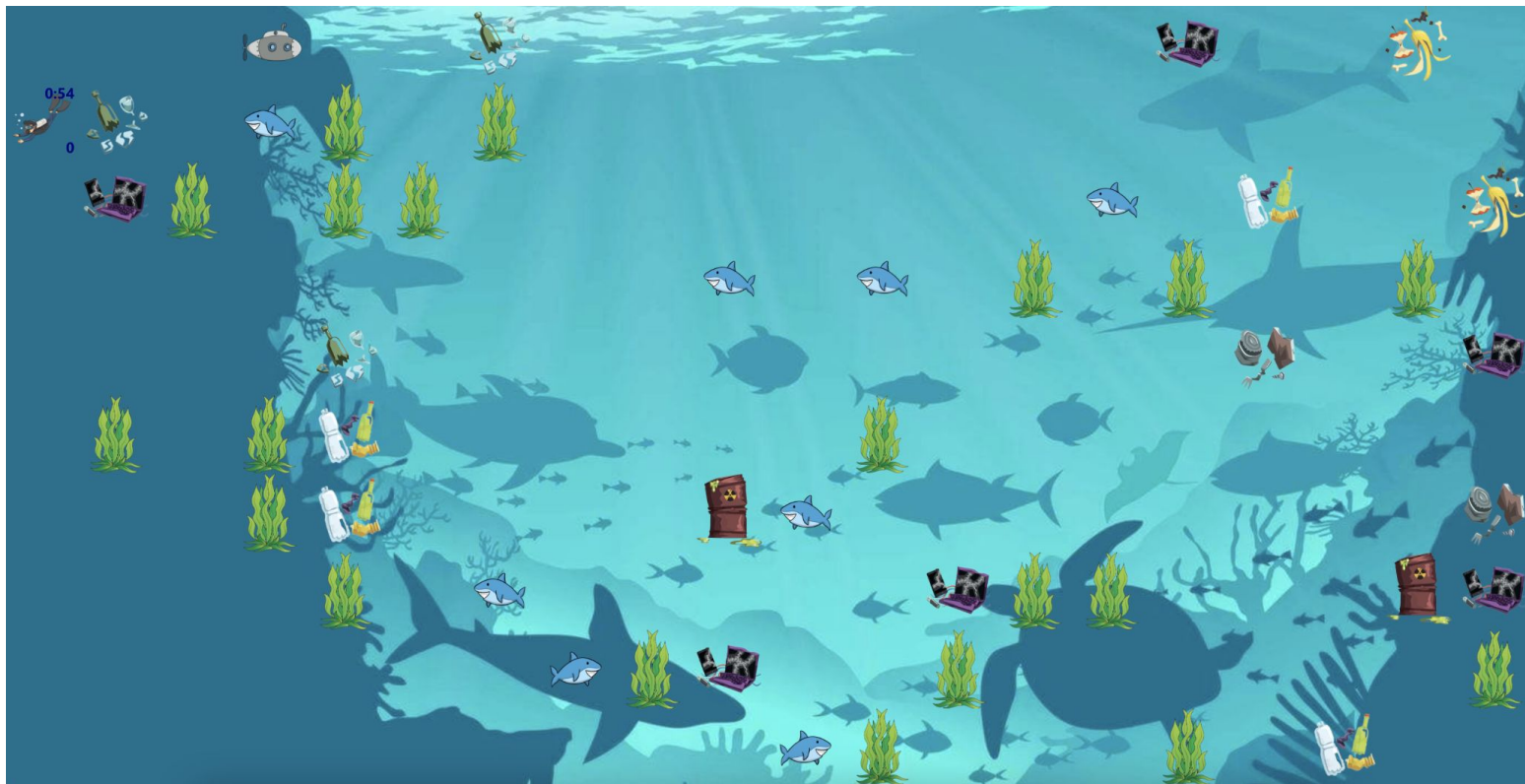
¡Vital para el análisis  
de datos!

# QUÉ SON LAS SHINY APPS

Es un entorno de desarrollo web en donde se pueden crear desde páginas web, hasta aplicaciones.

Ejemplos:

# SHARK ATTACK



# 69 LOVE SONGS

69 Love Songs

Intro

How F&#gt;king Romantic?

The Book of Love

Love, or Nothing At All

Love and Trouble

Making Me Blue

## 69 Love Songs: A Lyrical Analysis

*"Don't fall in love with me yet, we've only recently met..."* 1.1 - Absolutely Cuckoo

In 1999, the indie-pop band **Magnetic Fields** released the album **69 Love Songs**. Conceived and written by frontman, Stephin Merritt, it is a three-volume concept album containing (yep, you guessed it) 69 love songs.

Merritt has described the album as "...not remotely an album about love. It's an album about love songs..."

So, what does Stephin sing about when he sings about love (songs)?

Play with this interactive tool and find out!



Built by **committedtotape** using the power of Rstudio and Shiny.

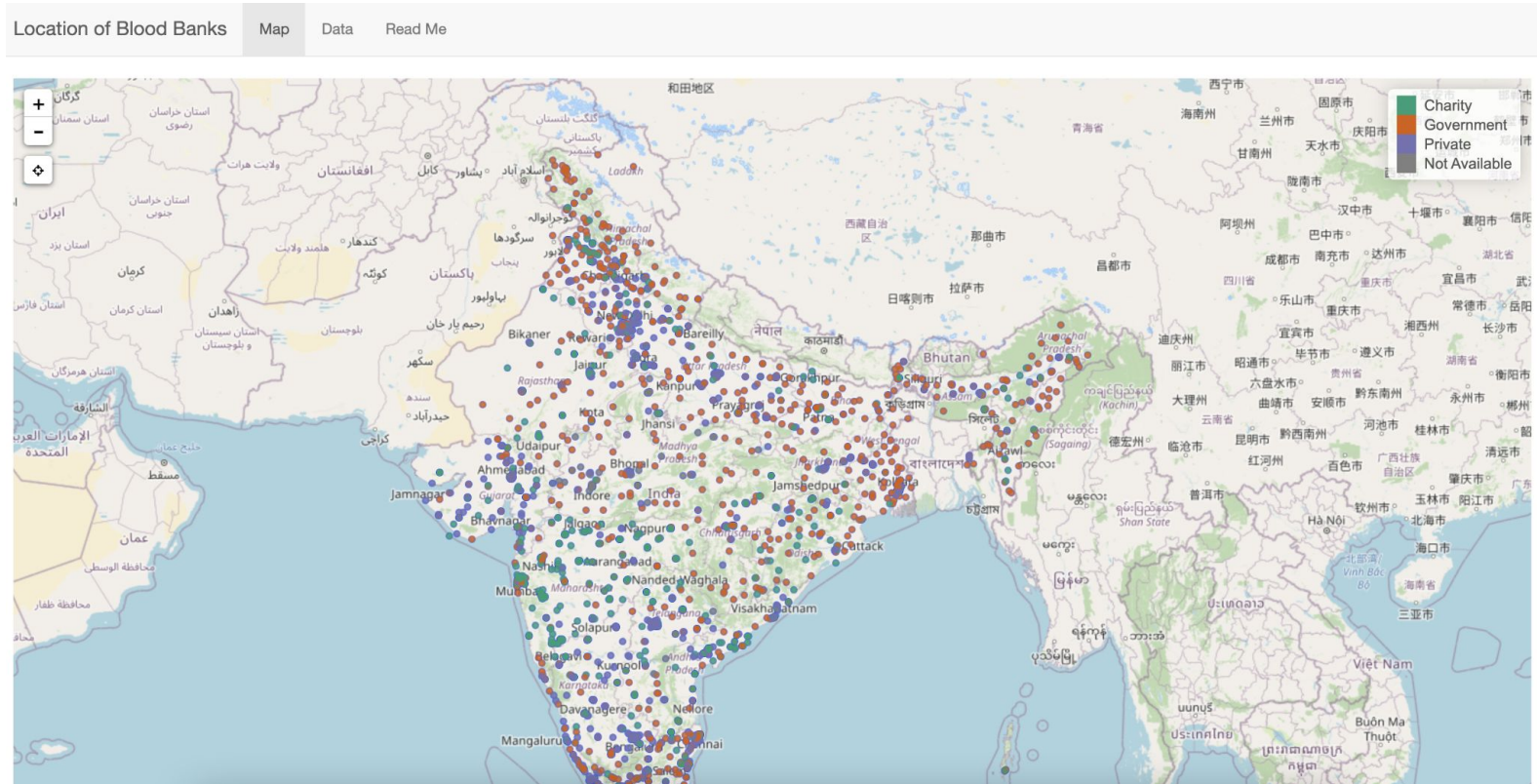
R Packages: tidyverse, tidytext, wordcloud2, tidygraph, vizNetwork, glue.

Sources: **genius.com** for lyrics, **wikipedia** for singers.

Fuente: <https://shiny.rstudio.com/gallery/magnetic-fields.html>



# UBICACIÓN DE BANCOS DE SANGRE EN LA INDIA



Fuente: <https://shiny.rstudio.com/gallery/india-blood-banks.html>

# TODOS LOS EJEMPLOS

Shiny from R Studio

Get Started

Gallery

Articles

App Stories

Reference

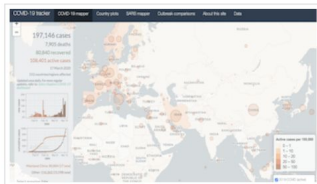
Deploy

Help

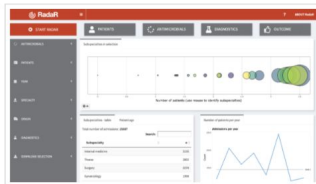
Contribute



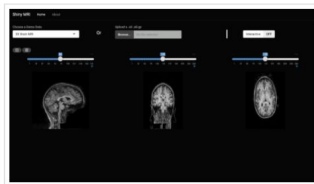
## Life sciences



COVID-19 tracker



Exploring large hospital data for better use of antimicrobials



ShinyMRI - View MRI images in Shiny



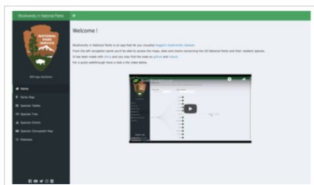
Nutrition Calculator - calculate nutrition for recipes



A/B Testing Sample Size Calculator



ctmweb, a web app to analysis Animal tracking data



Visualizing Biodiversity in National Parks data

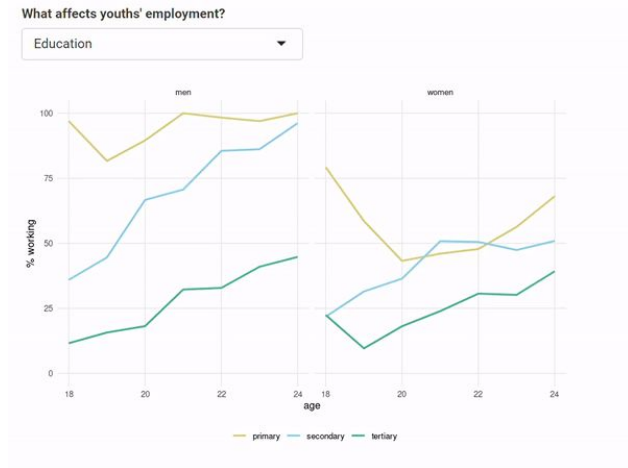


ExPanD: Explore Your Data Interactively

<https://shiny.rstudio.com/gallery/#user-showcase>

# ¿QUÉ TIENEN EN COMÚN ESTAS SHINY APPS?

- Fueron hechas con código.
- Usa varios paquetes del entorno R.
- Facilita la interacción  
datos - personas.

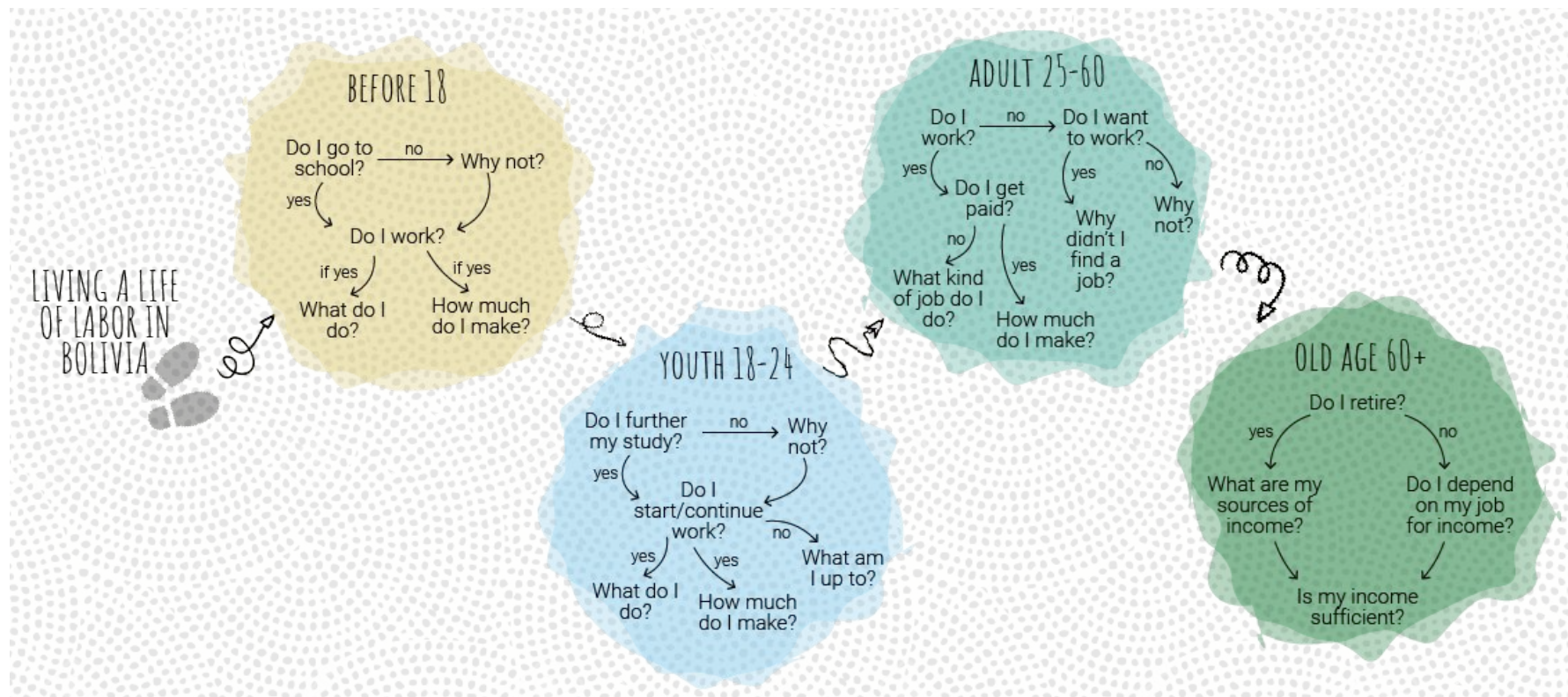


# VENTAJAS DE UNA SHINY APP

- Aprovecha los conocimientos que ya tenías de R. Por ello, es fácil de usar.
- Te ayuda a presentar e **interactuar** con los datos de diversas maneras: Tablas, **visualizaciones**, texto, etc.
- Es muy fácil de compartir (un link a [shinyapps.io](https://shinyapps.io) y es todo).
- Es un entorno muy customizable que da resultados atractivos.

¿PREGUNTAS?

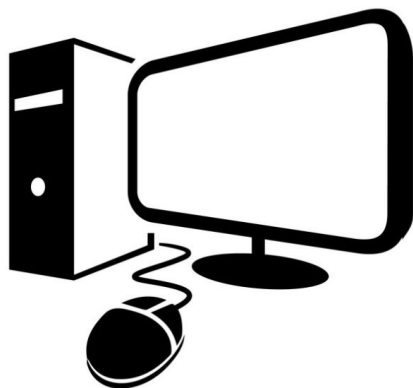
# NUESTRO EJEMPLO PERSONAL DE SHINY APP:



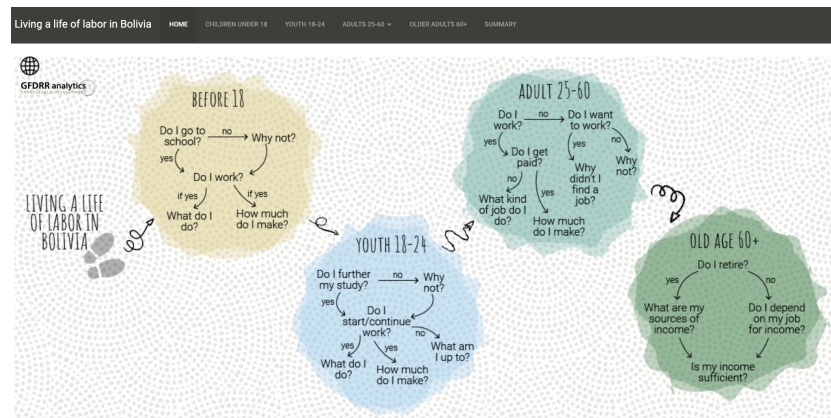
¿PREGUNTAS?



# ARQUITECTURA DE UNA SHINY APP



Servidor con instrucciones



Interface de usuario

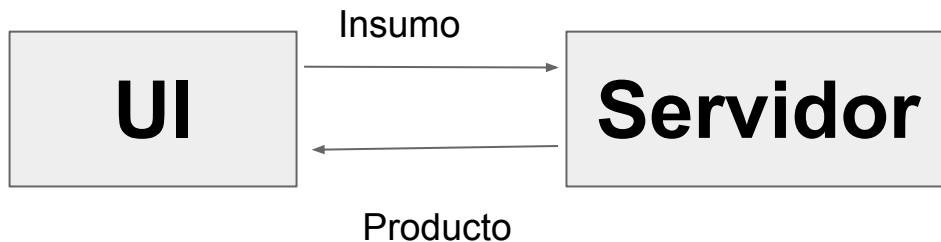


# ARQUITECTURA DE UNA SHINY APP:

Toda shiny app tiene 3 elementos principales:

1. La interface de usuario (User Interface o UI en inglés).
2. El servidor (server).
3. La función shinyApp, que hace interactuar 1 y 2.

1. Selecciona un insumo



2. Servidor ve el insumo seleccionado.

4. La interface exhibe el resultado/producto.

3. Servidor analiza/procesa insumo y bota un resultado.

# PLANTILLA DE UNA SHINY APP (LA MÁS BÁSICA).

```
library(shiny)
```

```
ui <- fluidPage()
```

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

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

# INPUTS

Los inputs son valores que los usuarios pueden proveer a R para que este lo procese y nos dé un resultado.

Se introducen mediante una función de Inputs.

# TUTORIAL : INPUTS

```
sliderInput(  
  inputId = 'num',  
  label = "Choose a number",  
  value = 25, min = 1, max = 100  
)
```

\*\*Cuidado con el CTRL + C porque las “” son diferentes a las que usa R!

# TUTORIAL: INPUTS, CÓMO SE VERÍA EN EL CÓDIGO

```
library(shiny)
ui <- fluidPage(
  sliderInput(inputId = 'num',
    label = "Choose a number",
    value = 25, min = 1, max = 100)
)
server <- function(input, output){}
shinyApp(ui = ui, server = server)
```

# TUTORIAL...

## Basic widgets

### Buttons

Action

Submit

### Single checkbox

☒ Choice A

### Checkbox group

☒ Choice 1

☐ Choice 2

☐ Choice 3

### Date input

2014-01-01

### Date range

2017-06-21

to

2017-06-21

### File input

Browse...

No file selected

### Help text

Note: help text isn't a true widget, but it provides an easy way to add text to accompany other widgets.

### Numeric input

1

### Radio buttons

☒ Choice 1

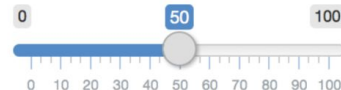
☐ Choice 2

☐ Choice 3

### Select box

Choice 1

### Sliders



### Text input

Enter text...

# TUTORIAL...

## Buttons

Action

Submit

`actionButton()`  
`submitButton()`

## Single checkbox

☒ Choice A

`checkboxInput()`

## Checkbox group

☒ Choice 1  
☐ Choice 2  
☐ Choice 3

`checkboxGroupInput()` `dateInput()`

## Date input

2014-01-01

## Date range

2014-01-24 to 2014-01-24

`dateRangeInput()`

## File input

Choose File No file chosen

`fileInput()`

## Numeric input

1

`numericInput()`

## Password Input

.....

`passwordInput()`

## Radio buttons

☒ Choice 1  
☐ Choice 2  
☐ Choice 3

`radioButtons()`

## Select box

Choice 1

`selectInput()`

## Sliders

0 50 100  
0 25 75 100

`sliderInput()`

## Text input

Enter text...

`textInput()`

# OUTPUTS

Objetos de R que el usuario puede ver:

Function	Inserts
<code>dataTableOutput()</code>	an interactive table
<code>htmlOutput()</code>	raw HTML
<code>imageOutput()</code>	image
<code>plotOutput()</code>	plot
<code>tableOutput()</code>	table
<code>textOutput()</code>	text
<code>uiOutput()</code>	a Shiny UI element
<code>verbatimTextOutput()</code>	text



# TUTORIAL: OUTPUTS, CÓMO SE VERÍA EN EL CÓDIGO

```
library(shiny)

ui <- fluidPage(
  sliderInput(inputId = 'num',
    label = "Choose a number",
    value = 25, min = 1, max = 100),
  plotOutput("hist")
)

server <- function(input, output){}

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

# RECAPITULACIÓN

- Empezar todas las apps con la plantilla.
- Agregar elementos al `fluidPage` como argumentos.
- Agregar elementos input con las funciones `Input()`
- Agregar elementos output con las funciones `Output()`
- Usar la función `server` para convertir inputs a outputs.

# LA FUNCIÓN SERVER

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

# LA FUNCIÓN SERVER: 1) CÓMO GUARDAR LOS OBJETOS OUTPUT

```
server <- function(input, output){  
  output$hist  
}
```

siempre guardar un objeto output con **output\$**

# LA FUNCIÓN SERVER: 1) CÓMO GUARDAR LOS OBJETOS OUTPUT

```
output$hist  
plotOutput("hist")
```

# LA FUNCIÓN SERVER: 2) LOS OBJETOS SE VISUALIZARÁN CON LA FUNCIÓN RENDER

function	creates
<code>renderDataTable()</code>	An interactive table <small>(from a data frame, matrix, or other table-like structure)</small>
<code>renderImage()</code>	An image (saved as a link to a source file)
<code>renderPlot()</code>	A plot
<code>renderPrint()</code>	A code block of printed output
<code>renderTable()</code>	A table <small>(from a data frame, matrix, or other table-like structure)</small>
<code>renderText()</code>	A character string
<code>renderUI()</code>	a Shiny UI element

LA FUNCIÓN SERVER: 2) LOS OBJETOS SE VISUALIZARÁN CON LA FUNCIÓN RENDER

```
server <- function(input, output){  
  output$hist <- renderPlot({hist(rnorm(100))  
})  
  
}
```

## LA FUNCIÓN SERVER: 3) CÓMO ACCEDER A LOS VALORES DE LOS INPUTS

```
server <- function(input, output){  
  output$hist <- renderPlot({hist(input$num)  
})  
}
```

Usas los valores input como **input\$**



# CÓMO SE RELACIONAN INPUTS Y OUTPUTS:

```
sliderInput(inputId = 'num', ... )
```

Es el mismo ID de

```
input$num
```

¿PREGUNTAS?

MUCHAS GRACIAS