Build Your Own Interactive Dashboard with R Shiny

By: Erika Siregar

Presented on TalksOn #34

8 April 2021

Let's get acquainted first

Please download the following materials: http://bit.ly/iykra34

Hello, my name is Erika Siregar

1. Cofounder **R-Ladies Jakarta**
2. Coordinator **Jakarta Machine Learning (JML)**
3. Connect with me:
   a. Email: erika.mukhlisina@gmail.com
   b. Twitter: @erikaris
   c. IG: @erikaris15
   d. GitHub: https://github.com/erikaris

A Little Bit about R-Ladies Jakarta

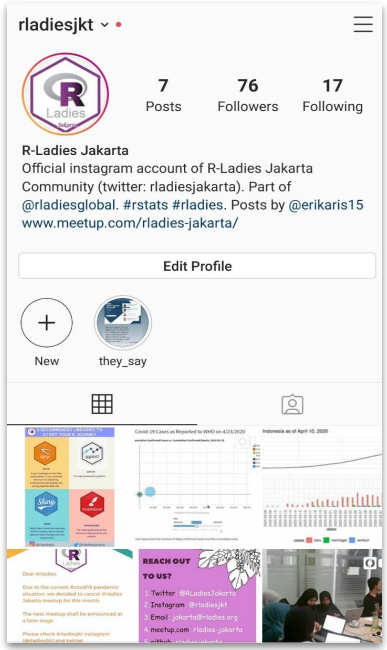
- **Komunitas belajar bersama** bahasa R untuk mem-promote gender diversities
  - Prioritas: perempuan
  - Laki-laki boleh ikut
  - Accepting all levels of expertise
- Established: July 2019
- 13 meetup: from intro to visualization and data analytics
  - hands-on
  - Check our github: https://github.com/rladiesjakarta

A Sneak Peek of Our Meetups

More about Us

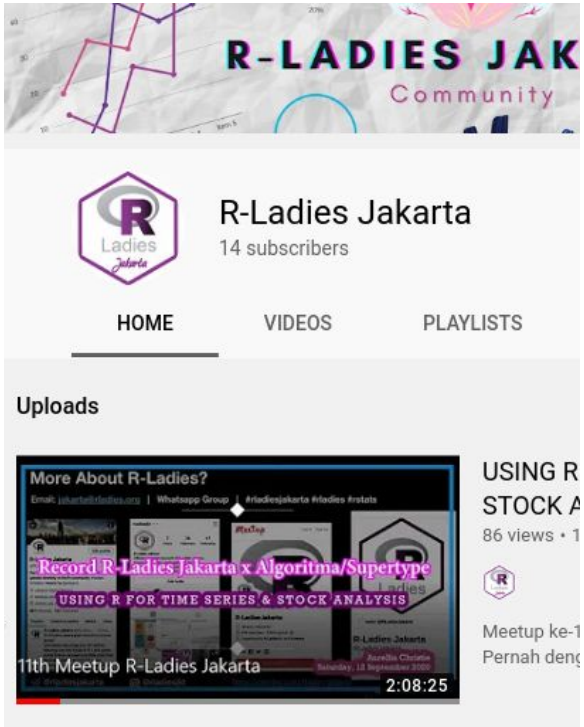
All materials & recording from previous meetups

@rladiesjakarta



@rladiesjkt

@rladiesjakarta



R-Ladies Jakarta
http://bit.ly/rladiesjkt_ytb

Events & R tips

In progress

- Landing Page: https://rladiesjakarta.github.io
- Join us, register on:
  - https://rladiesjakarta.github.io/#/registration

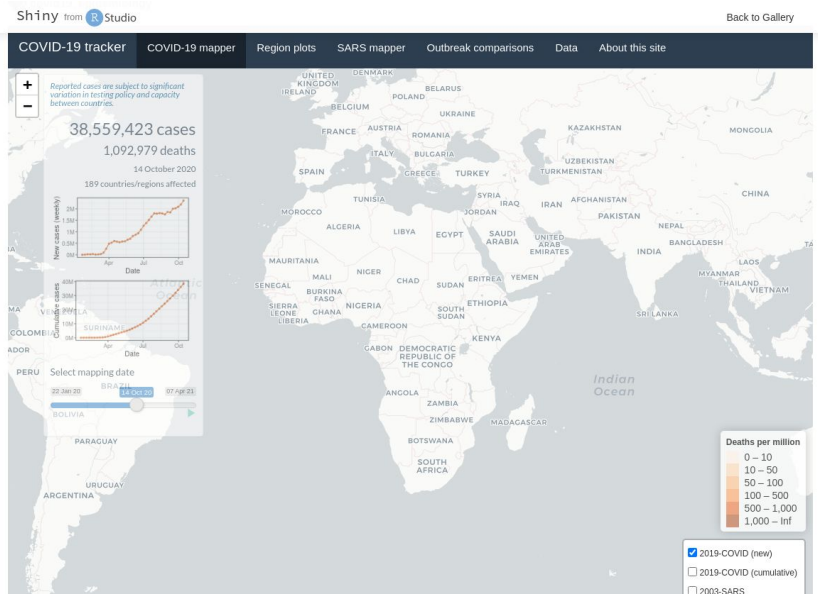
Now, let's get started with Shiny

Self Check

1. Pernah menggunakan R?
2. Pernah menggunakan RStudio?
3. Pernah dengar tentang R Shiny?
4. Experience with (simple) web programming?

Visualization and Dashboarding

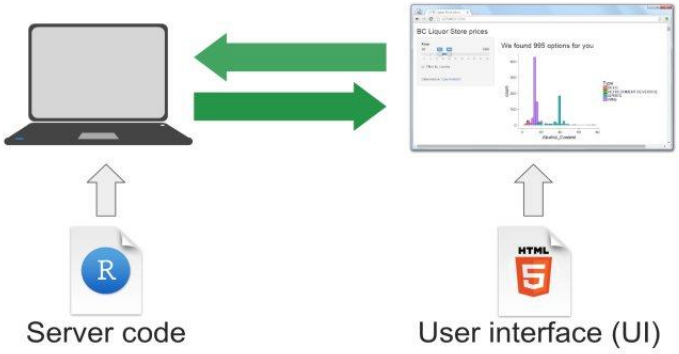
1. We have data, how to get insights? VISUALIZATION
2. We have a **plethora** of data, how to get **multiperspective** insights? DASHBOARD.
   a. > 1 visualization
   b. Snapshot of recent numbers (kondisi angka terkini)
   c. User interaction → widgets



https://shiny.rstudio.com/gallery/covid19-tracker.html

What is Shiny?

1. an **R package** for building an interactive web apps straight from R.
2. Built on bootstrap → responsive page
3. can be hosted as a standalone apps on a webpage ; or embed them in R Markdown documents ; or **build dashboards**.



R code → convert HTML → liatkan inspect element.

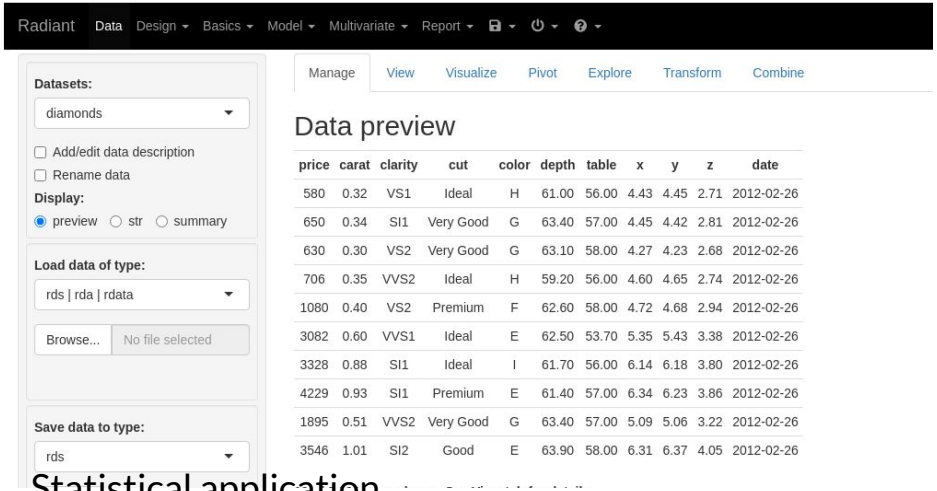
More details: https://shiny.rstudio.com/

Why Shiny?

- Interactivity & animation
- Free
- Open source (no license required) → https://github.com/rstudio/shiny
- Extensible & highly customized
  - Web-based → customize with **CSS**, htmlwidgets, and JavaScript (plotly, d3, etc).
  - Available extension libraries → make your dashboard fancier. → shinyflexdashboard, shinywidgets, etc → https://awesomeopensource.com/project/nanxstats/awesome-shiny-extensions
- R-based → get the computational power of R.

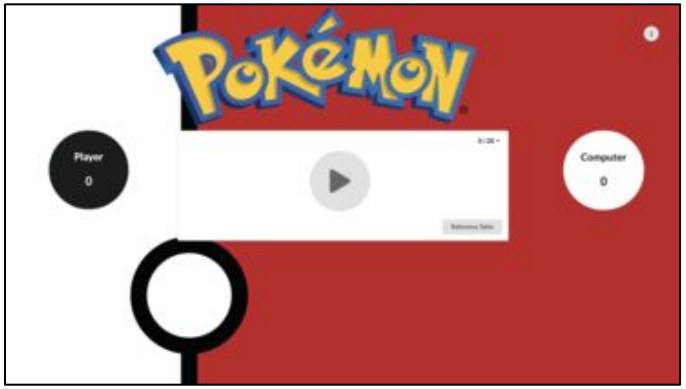
What can We Do with Shiny

dashboard

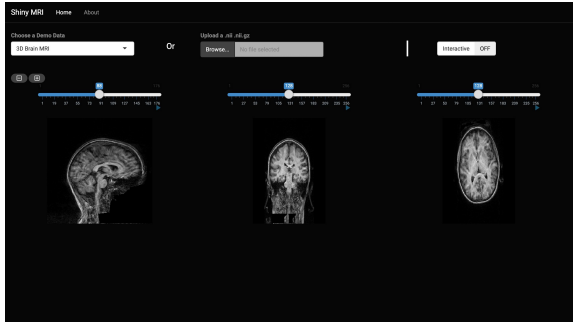


price	carat	clarity	cut	color	depth	table	x	y	z	date
580	0.32	VS1	Ideal	H	61.00	56.00	4.43	4.45	2.71	2012-02-26
650	0.34	SI1	Very Good	G	63.40	57.00	4.45	4.42	2.81	2012-02-26
630	0.30	VS2	Very Good	G	63.10	58.00	4.27	4.23	2.68	2012-02-26
706	0.35	VVS2	Ideal	H	59.20	56.00	4.60	4.65	2.74	2012-02-26
1080	0.40	VS2	Premium	F	62.60	58.00	4.72	4.68	2.94	2012-02-26
3082	0.60	VVS1	Ideal	E	62.50	53.70	5.35	5.43	3.38	2012-02-26
3328	0.88	SI1	Ideal	I	61.70	56.00	6.14	6.18	3.80	2012-02-26
4229	0.93	SI1	Premium	E	61.40	57.00	6.34	6.23	3.86	2012-02-26
1895	0.51	VVS2	Very Good	G	63.40	57.00	5.09	5.06	3.22	2012-02-26
3546	1.01	SI2	Good	E	63.90	58.00	6.31	6.37	4.05	2012-02-26

Statistical application



game



MRI visualization apps

Source:
https://shiny.rstudio.com/gallery/

How to get Started with Shiny?

1. Get the skill
   a. R → must
   b. Web-programming (css & javascript) → nice to have.
2. Install library → install.packages(shiny)
3. Load library → library(shiny)
4. Start building your dashboard.

The Components of Shiny

1. UI → frontend
   a. Input
   b. Output

   } layout

2. Server → backend → where the logic of the app is implemented (calculation, aggregation, etc.)

```
# Load the shiny package                    1
library(shiny)

# Define UI for the application
ui <- fluidPage(
  # Add the text "Shiny is fun"              2
  "nyobain shiny untuk pertama kalinya"
)

# Define the server logic
server <- function(input, output) {}        3

# Run the application
shinyApp(ui = ui, server = server)          4
```

UI & server are combined with shinyApp()

Let's give it a try: just simple text

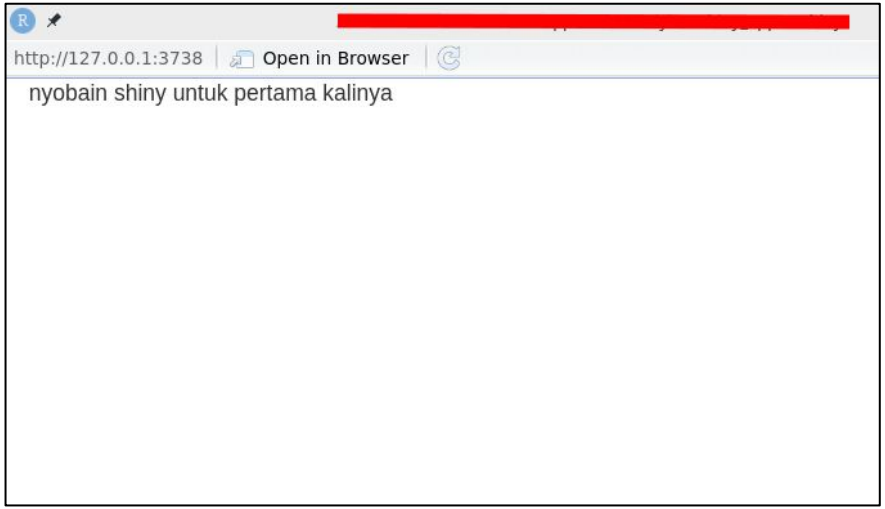
```r
# Load the shiny package
library(shiny)

# Define UI for the application
ui <- fluidPage(
  # Add the text "Shiny is fun"
  "nyobain shiny untuk pertama kalinya"
)

# Define the server logic
server <- function(input, output) {}

# Run the application
shinyApp(ui = ui, server = server)
```

1

2

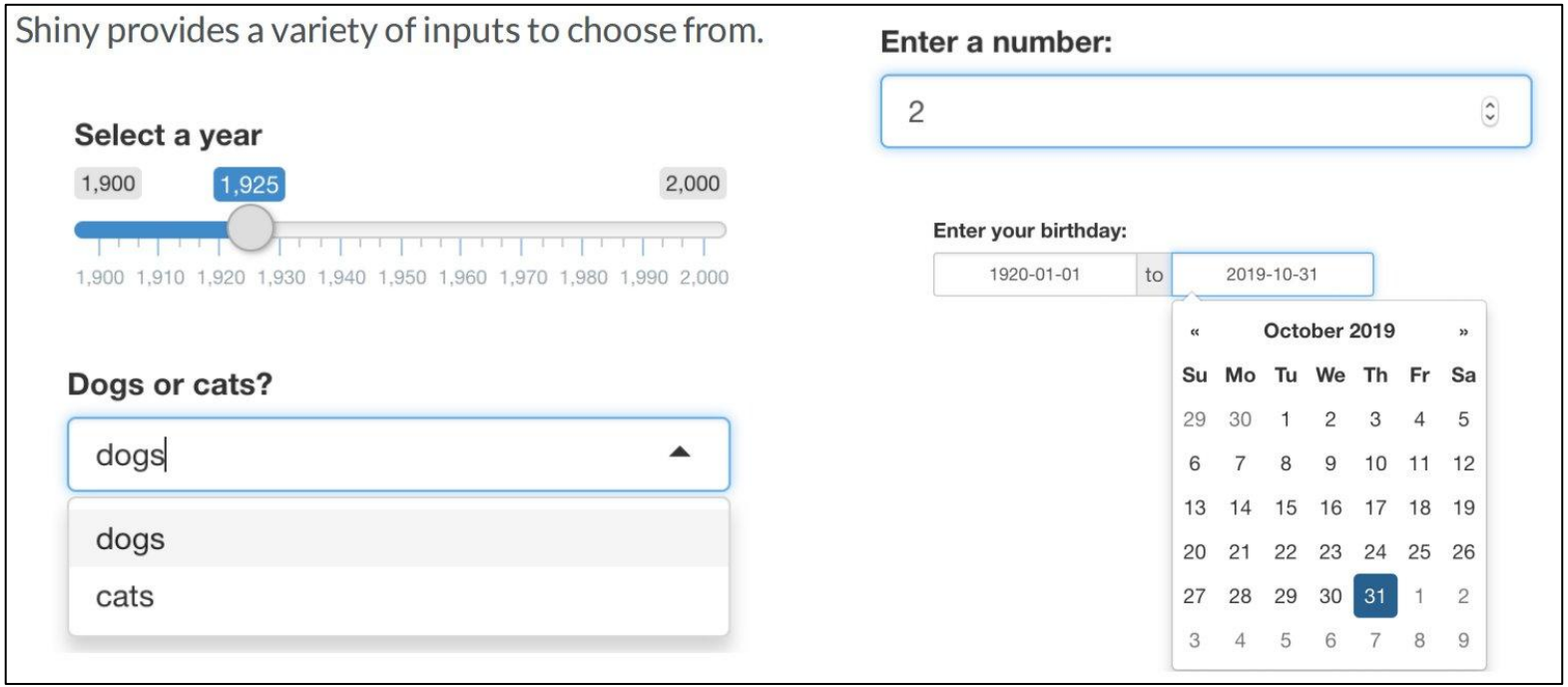
3

4



The UI Component: Input

- See the complete list on Shiny Cheatsheet
- Ciri sebuah input: *xxxInput*.



The UI Component: Output

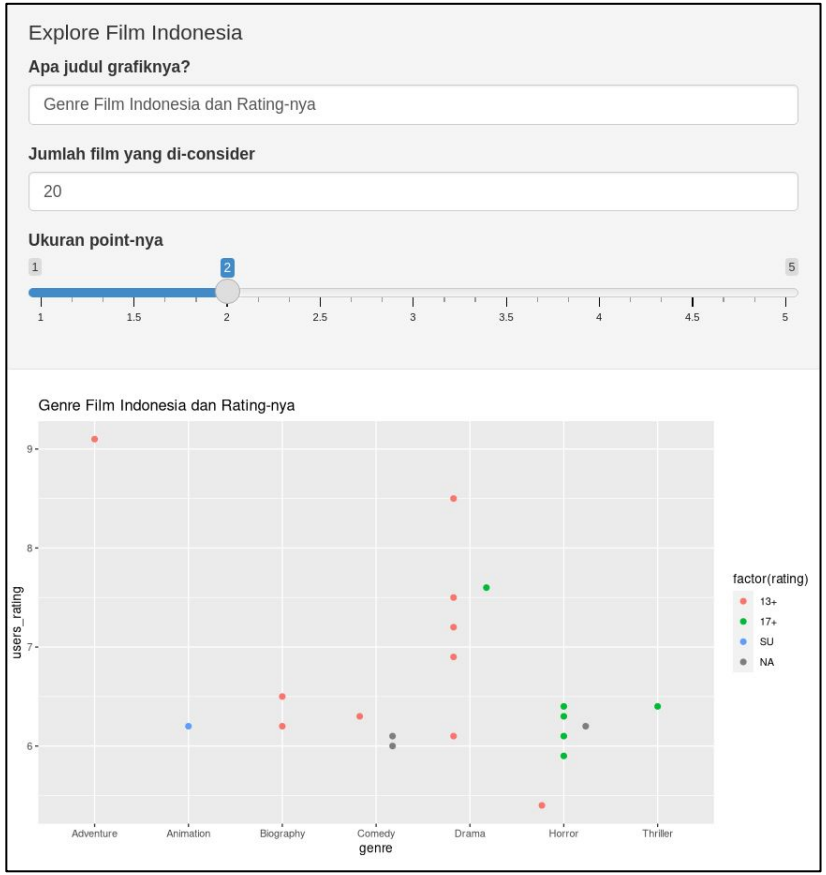
- See the complete list on Shiny Cheatsheet
- Ciri sebuah input: *xxxOutput*.
- Example: plotOutput, tableOutput, textOutput

How to connect input and output?

1. Each input (xxxInput) & output (xxxOutput) has an id
2. Connect them through this ids in the server using renderxxx.

```
# Load the shiny package
library(shiny)
library(ggplot2)

indonesian_movies <- read_csv("indonesian_movies.csv")

# Define UI for the application
ui <- fluidPage(
  # Add a sidebar layout to the application
  sidebarLayout(
    # Add a sidebar panel around the text and inputs
    sidebarPanel(
      h4("Explore Film Indonesia"),
      textInput("title", "Apa judul grafiknya?", "Genre Film Indonesia dan
Rating-nya"),
      numericInput("num", "Jumlah film yang di-consider", 20, 1, nrow
(indonesian_movies)),
      sliderInput("size", "Ukuran point-nya", 1, 5, 2, 0.5)
    ),
    # Add a main panel around the plot and table
    mainPanel(
      plotOutput("plot")
)))

# Define the server logic
server <- function(input, output) {
  output$plot <- renderPlot({
    ggplot(indonesian_movies[1:input$num, ], aes(x = genre, y =
users_rating))+
      geom_point(position = position_dodge(width = 0.7), size = input$size,
aes(colour = factor(rating))) +
      labs(title = input$title)
  })
}

# Run the application
shinyApp(ui = ui, server = server)
```

Put it in a practice

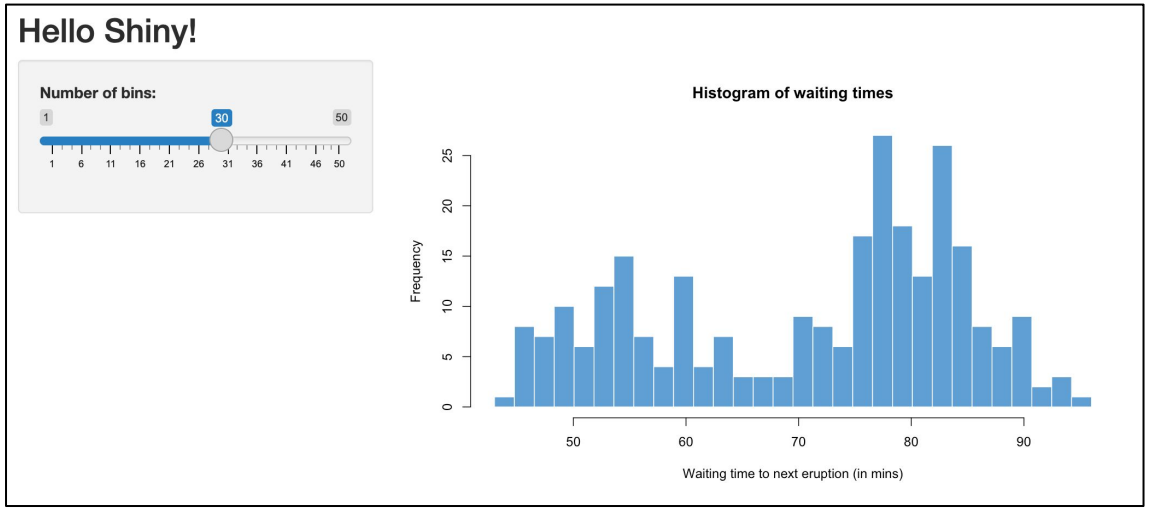



```r
# Load the shiny package
library(shiny)
library(ggplot2)

indonesian_movies <- read_csv("indonesian_movies.csv")

# Define UI for the application
ui <- fluidPage(
  # Add a sidebar layout to the application
  sidebarLayout(
    # Add a sidebar panel around the text and inputs
    sidebarPanel(
      h4("Explore Film Indonesia"),
      textInput("title", "Apa judul grafiknya?", "Genre Film Indonesia dan
Rating-nya"),
      numericInput("num", "Jumlah film yang di-consider", 20, 1, nrow
(indonesian_movies)),
      sliderInput("size", "Ukuran point-nya", 1, 5, 2, 0.5)
    ),
    # Add a main panel around the plot and table
    mainPanel(
      plotOutput("plot")
)))

# Define the server logic
server <- function(input, output) {
  output$plot <- renderPlot({
    ggplot(indonesian_movies[1:input$num, ], aes(x = genre, y =
users_rating))+
      geom_point(position = position_dodge(width = 0.7), size = input$size,
aes(colour = factor(rating))) +
      labs(title = input$title)
  })
}

# Run the application
shinyApp(ui = ui, server = server)
```

Summary of UI-Server relationship

No	Component	UI		Server
		Input	Output	Render
1	text	textInput(id, label, default_value)	textOutput(id)	output$id <- renderText({expr})
	numeric input	numericInput(id, label, default_value)	-	-
2	plot	-	plotOutput(id)	output$id <- renderPlot({expr})
3	dropdown	selectInput(id, label, selected, choices)	-	-
4	slider	sliderInput(id, label, min, max, value)	-	-
5	table	-	tableOutput(id)	output$id <- renderTable({expr})
6	checkbox	checkboxInput(id, label)	-	-
7	radiobutton	radioButtons(inputId, label, choices = NULL, selected = NULL)	-	-
8	colour	colourInput(id, label)	-	-
9	file	fileInput(id, label)	-	-

Shiny Layout: Sidebar



sidebarPanel

mainPanel

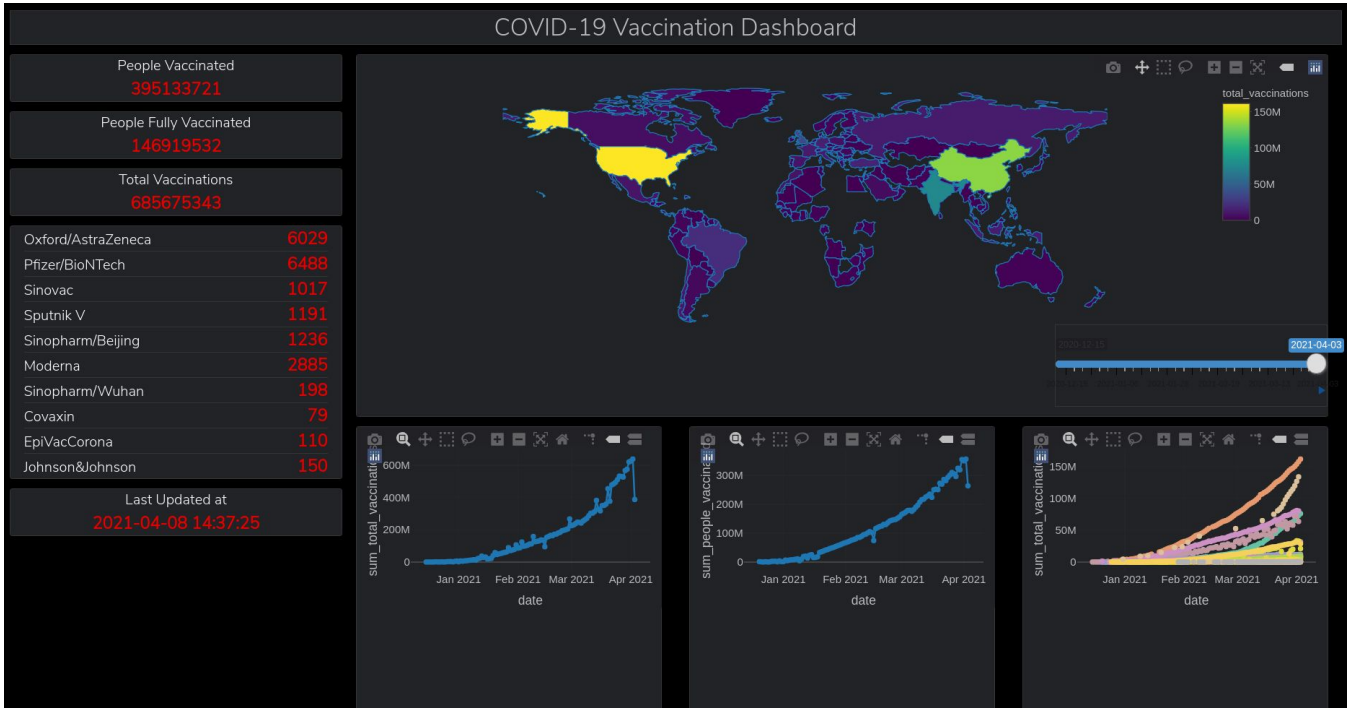
```
ui <- fluidPage(

  titlePanel("Hello Shiny!"),

  sidebarLayout(

    sidebarPanel(
      sliderInput(
        "bins", label = "Number of bins:",
        min = 1, value = 30, max = 50
      )
    ),

    mainPanel(
      plotOutput("distPlot")
    )
  )
)
```

Shiny Layout: Grid

For code, go check https://shiny.rstudio.com/articles/layout-guide.html

Demo Time!!!

Goals:



Data:

- https://ourworldindata.org/
- https://github.com/owid/covid-19-data/tree/master/public/data/vaccinations

Thank you .

Any question?