

Introduction to Shiny

BUILDING WEB APPLICATIONS WITH SHINY IN R



Ramnath Vaidyanathan

VP of Product Research

Introduction to Shiny

http://127.0.0.1:4397

Open in Browser



Publish

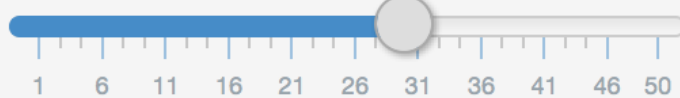
Hello Shiny!

Number of bins:

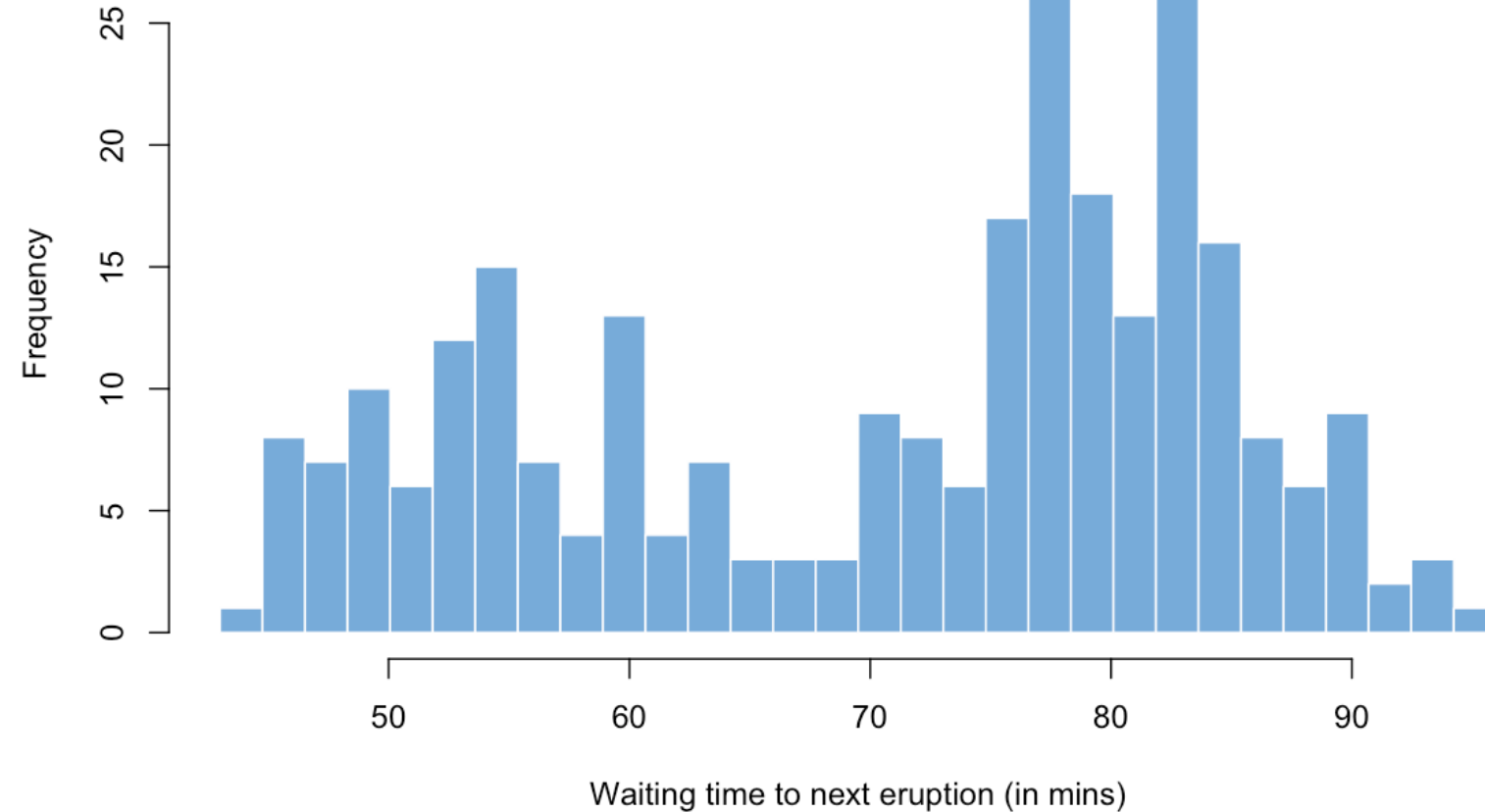
1

30

50



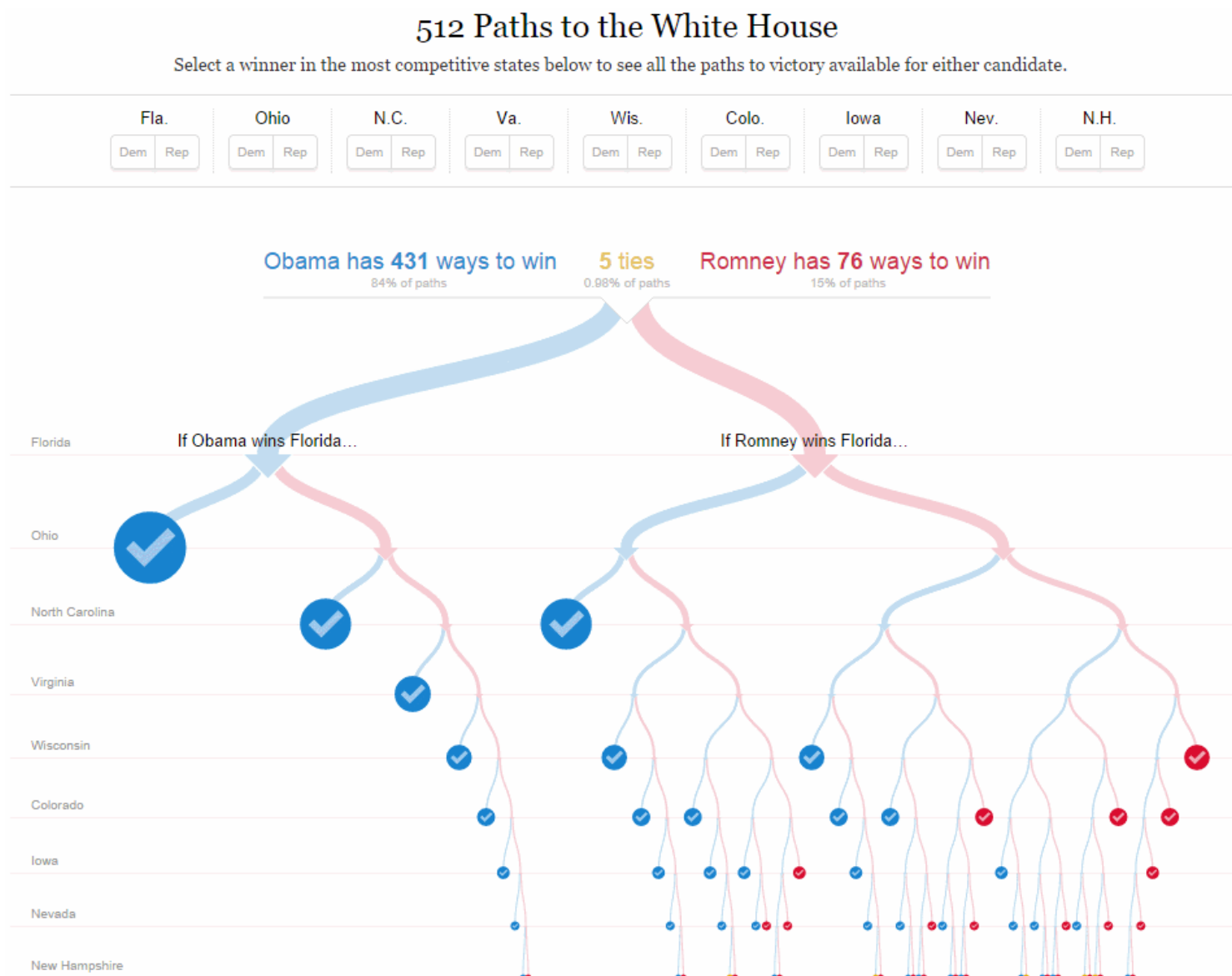
Histogram of waiting times



What is a web app?

- Updates based on user input/interaction
- Made up of UI & server

What is a web app?



- Displays paths to the White House for different presidential candidates.

What is a web app?

- DataCamp mobile app

ⓘ

×

1	Jan 2014	X
2	Jan 2015	E
3	Jan 2016	D
4	Feb 2014	X
5	Feb 2015	C
6	Feb 2016	C

What is the output of this code?

```
summarise(df, result = n_distinct(month))
```

Select the output

result

1 2

result

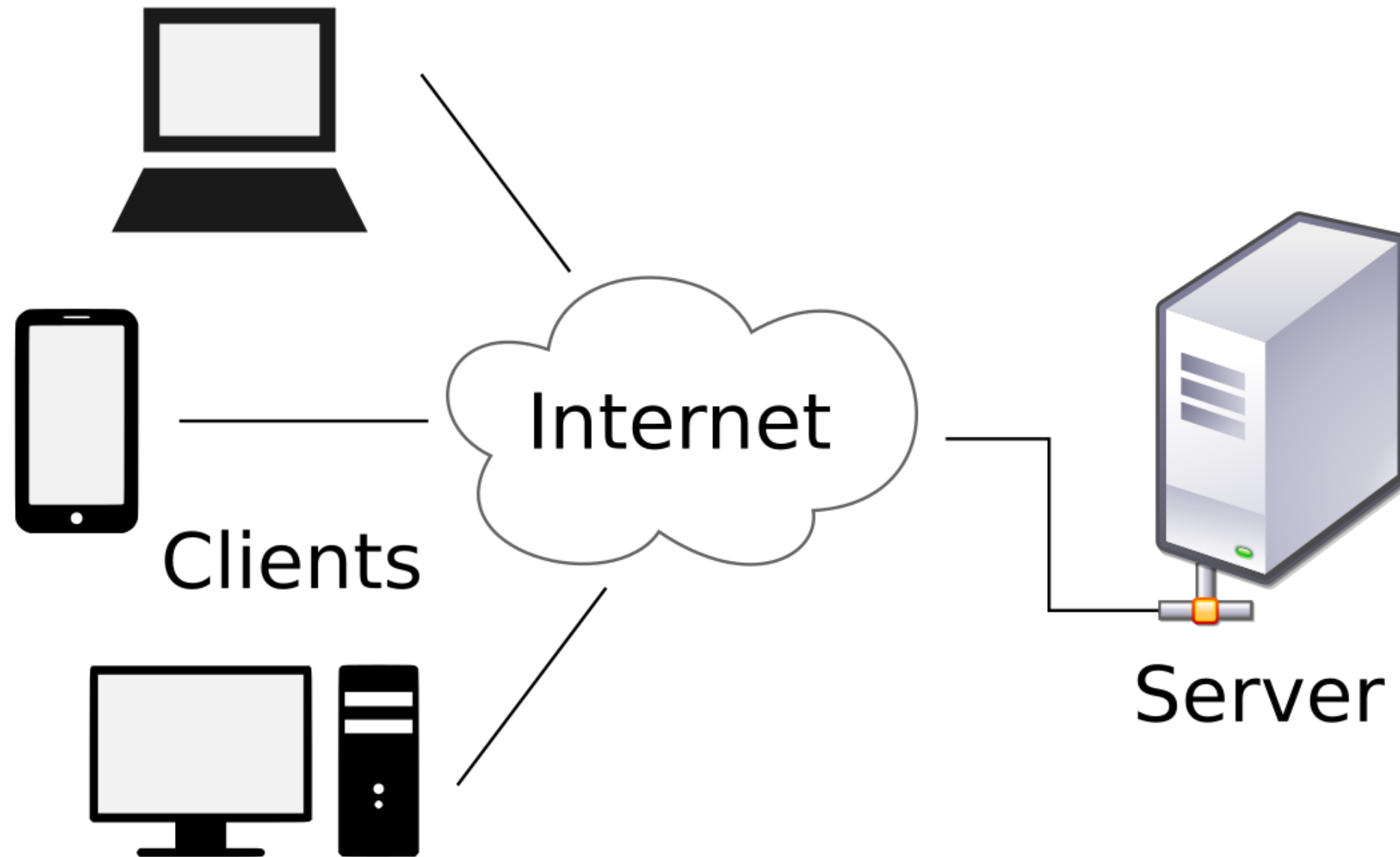
1 4

result

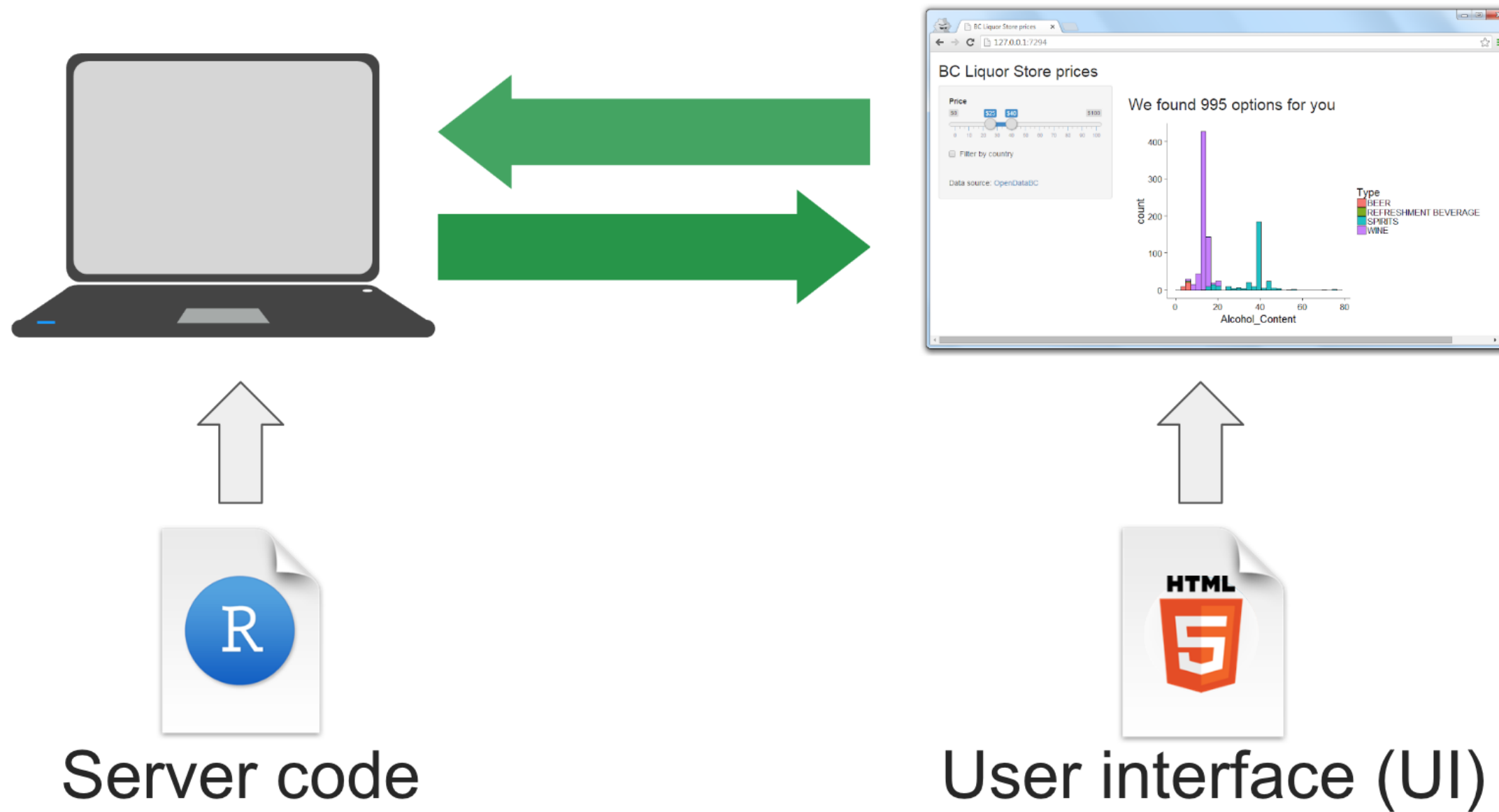
1 3

How does a web app work?

A web app is a thing that updates based on user input/interaction



What is Shiny?

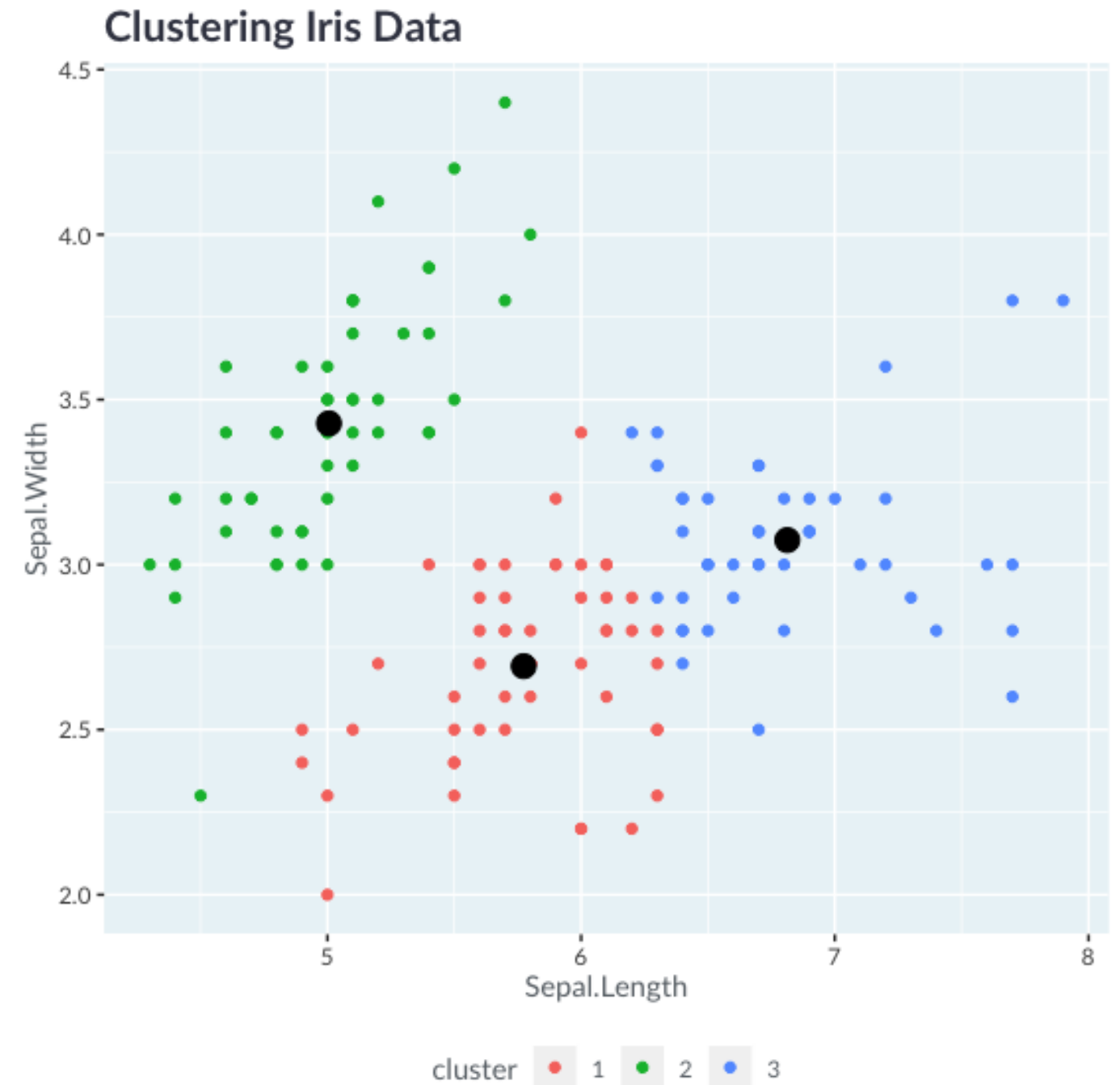


Why should data scientists build web apps?



Why should data scientists build web apps?

```
plot_kmeans(  
  data = iris,  
  x = 'Sepal.Length',  
  y = 'Sepal.Width',  
  nb_clusters = 3  
)
```



Why should data scientists build web apps?

```
library(shiny)
ui <- fluidPage(
  h1('K-Means Clustering App'),
  selectInput('x', 'Select x', names(iris), 'Sepal.Length'),
  selectInput('y', 'Select y', names(iris), 'Sepal.Width'),
  numericInput('nb_clusters', 'Select number of clusters', 3),
  plotly::plotlyOutput('kmeans_plot')
)

server <- function(input, output, session){
  output$kmeans_plot <- plotly::renderPlotly({
    plot_kmeans(iris, input$x, input$y, input$nb_clusters)
  })
}

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

Why should data scientists build web apps?

K-Means Clustering App

Select x

Sepal.Length

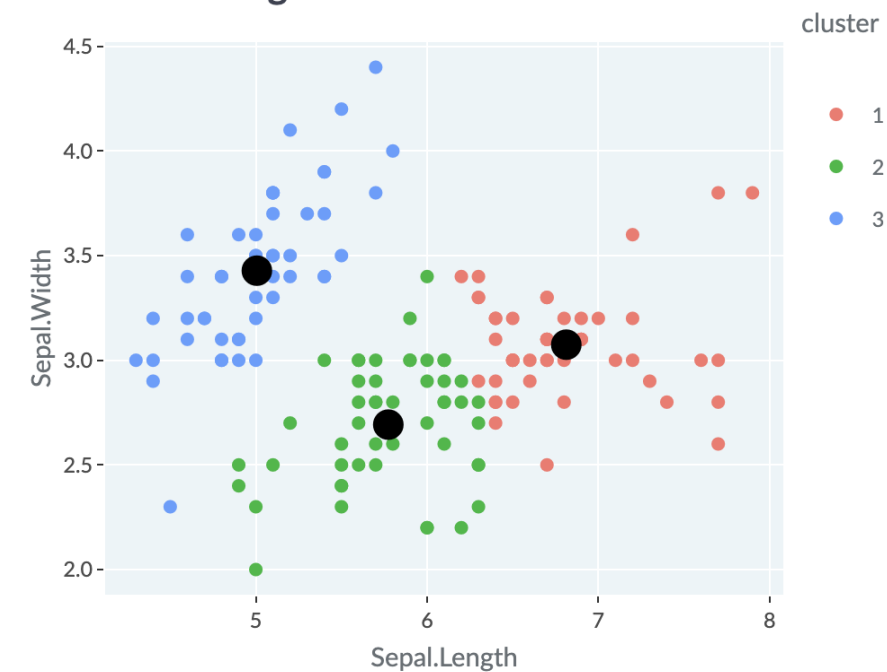
Select y

Sepal.Width

Select number of clusters

3

Clustering Iris Data



Let's practice!

BUILDING WEB APPLICATIONS WITH SHINY IN R

Build a "Hello, world" Shiny app

BUILDING WEB APPLICATIONS WITH SHINY IN R



Kaelen Medeiros
Data Scientist

Parts of a Shiny app

```
library(shiny)
ui <- fluidPage()
server <- function(input,
                   output,
                   session) {

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

- Load `shiny`
- Create the UI with a HTML function
- Define a custom function to create the server
- Run the app

Hello, world!!!

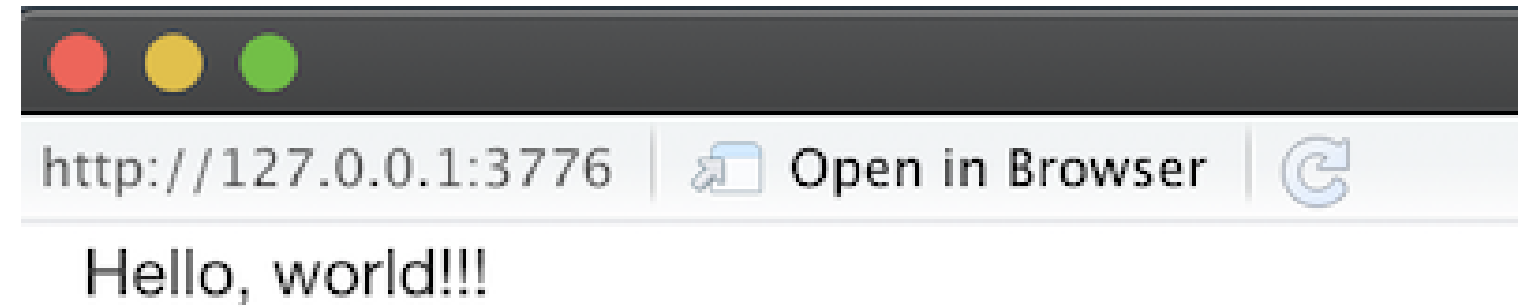
```
library(shiny)

ui <- fluidPage(
  "Hello, world!!!"
)

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

}

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



Ask a question (with an input!)

```
ui <- fluidPage(  
  textInput("name", "Enter a name:"),  
  textOutput("q")  
)  
server <- function(input, output) {  
  output$q <- renderText({  
    paste("Do you prefer dogs  
          or cats,",  
          input$name, "?")  
  })  
}
```

http://127.0.0.1:3776



Open in Browser



Enter a name:

Kaelen

Do you prefer dogs or cats, Kaelen ?

Let's practice!

BUILDING WEB APPLICATIONS WITH SHINY IN R

Build a babynames explorer Shiny app

BUILDING WEB APPLICATIONS WITH SHINY IN R



Ramnath Vaidyanathan

VP of Product Research

Sketch your app

Baby Name Explorer

Enter Name

David



Add inputs (UI)

```
ui <- fluidPage(  
  titlePanel("Baby Name Explorer"),  
  textInput('name', 'Enter Name', 'David')  
)
```

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

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

Baby Name Explorer

Enter Name

Add outputs (UI/server)

```
ui <- fluidPage(  
  titlePanel("Baby Name Explorer"),  
  textInput('name', 'Enter Name', 'David'),  
  plotOutput('trend')  
)
```

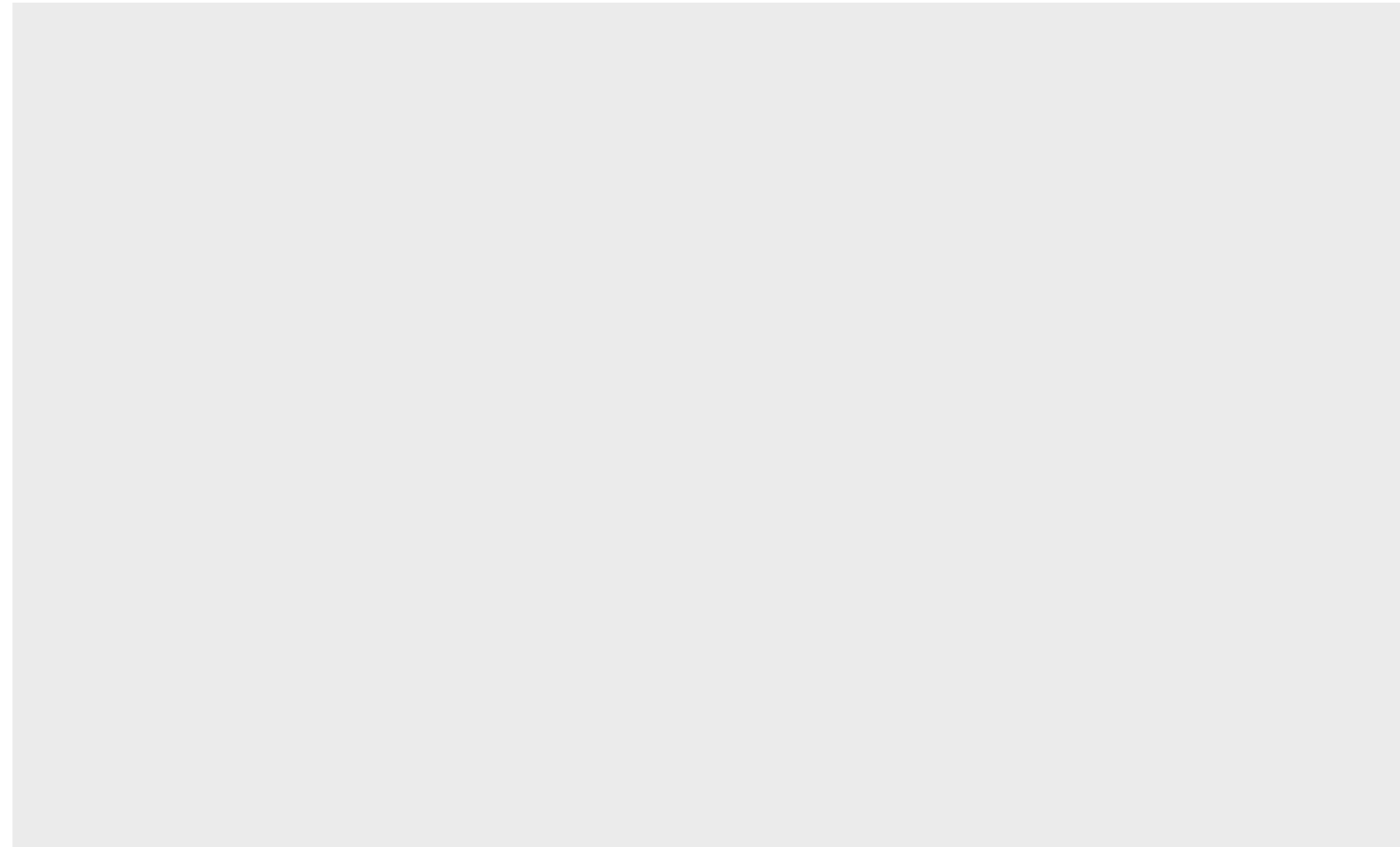
```
server <- function(input, output, session){  
  output$trend <- renderPlot({  
    ggplot()  
  })  
}
```

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

Add outputs (UI/server)

Baby Names Explorer

Enter Name



Update layout (UI)

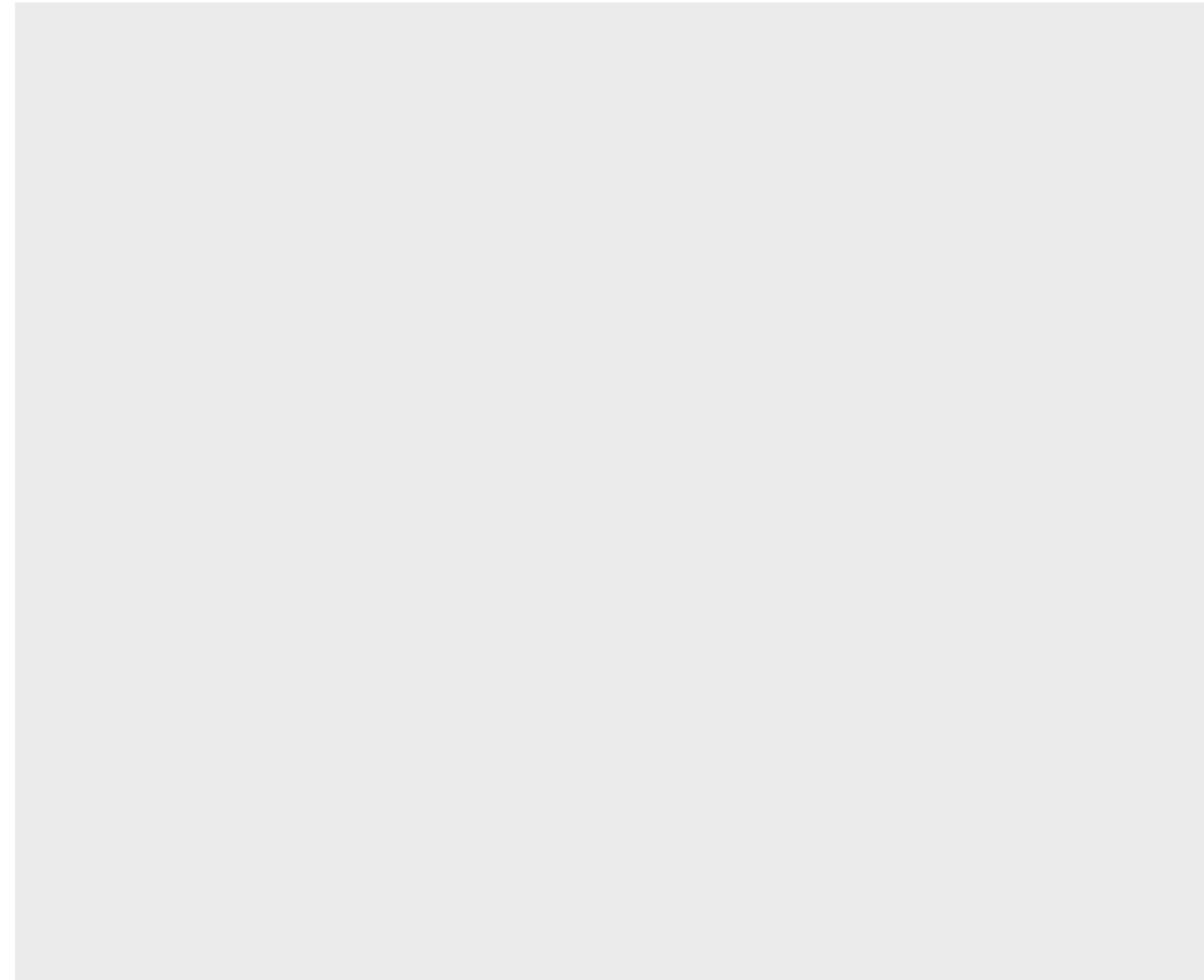
```
ui <- fluidPage(  
  titlePanel("Baby Name Explorer"),  
  sidebarLayout(  
    sidebarPanel(  
      textInput('name', 'Enter Name', 'David')  
    ),  
    mainPanel(  
      plotOutput('trend')  
    )  
  )  
)
```

```
server <- function(input, output, session){  
  output$trend <- renderPlot({ggplot()})  
}
```

Update layout (UI)

Baby Name Explorer

Enter Name



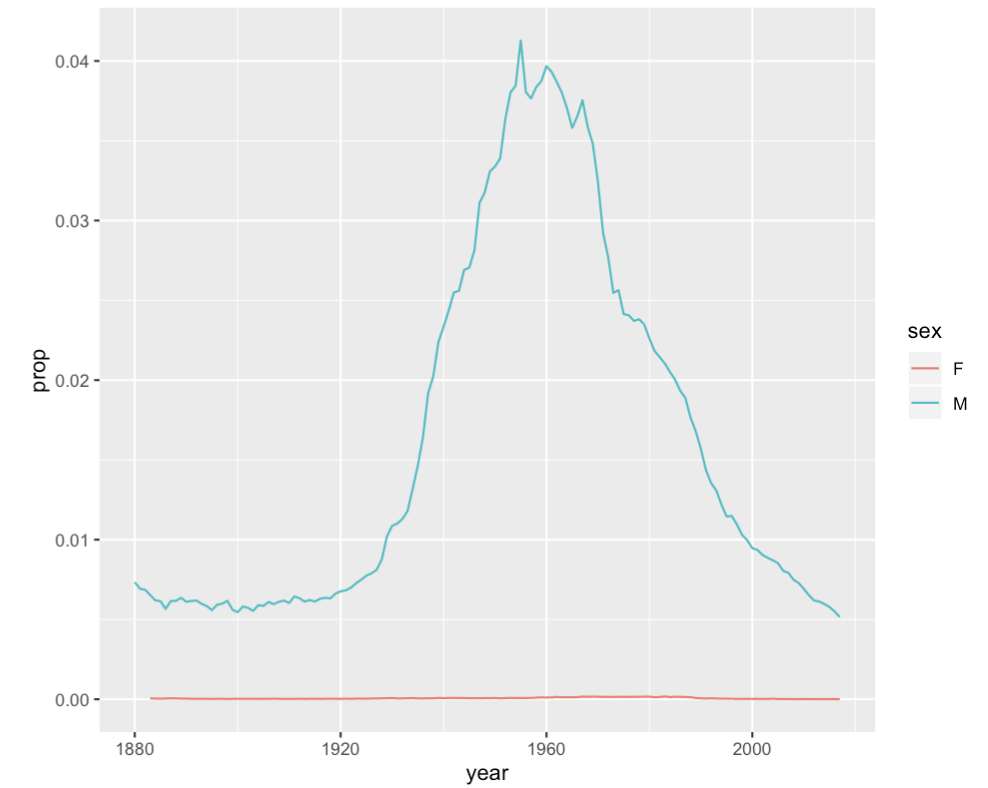
Update output (server)

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

```
server <- function(input, output, session){  
  output$trend <- renderPlot({  
    data_name <- subset(  
      babynames, name == input$name  
    )  
    ggplot(data_name) +  
      geom_line(  
        aes(x = year, y = prop, color = sex)  
      )  
  })  
}
```

Baby Name Explorer

Enter Name



Let's practice!

BUILDING WEB APPLICATIONS WITH SHINY IN R