



KMMI 2021

Eksplorasi dan Visualisasi Data

Pertemuan 10:
Visualisasi Menggunakan R Shiny

Sub CPMK

Mahasiswa mampu membuat grafik interaktif dengan R Shiny.

Pokok Bahasan

1. Menginstall dan memuat R shiny
2. Bagian dari skrip aplikasi Shiny serta kegunaannya masing-masing
3. Membuat aplikasi shiny dari contoh bawaan R
4. Membuat visualisasi interaktif dengan R Shiny

Visualisasi Interaktif dengan R Shiny

- ✓ **Paket Shiny** adalah paket R yang dapat membuat ui (*user interface*) interaktif dalam bentuk **aplikasi Web interaktif** yang dapat berubah berdasarkan input pengguna.
- ✓ **Aplikasi Shiny** memiliki halaman web yang **dikunjungi pengguna** dan di belakang halaman web terdapat komputer yang **melayani halaman** tersebut dengan menjalankan R.
- ✓ **Shiny** memiliki kumpulan fitur yang cukup banyak yang dapat digunakan untuk membuat aplikasi web yang kompleks.

Keuntungan Membuat Aplikasi Web

- ✓ Sebagian besar pengguna dapat mengakses visualisasi interaktif melalui browser web **tanpa menginstal paket R** dan **mengunduh data**.
- ✓ Pembuatan aplikasi web dengan paket Shiny **seluruhnya dilakukan di R**, tanpa menulis kode HTML atau Javascript.
- ✓ Meskipun demikian, dimungkinkan untuk menentukan UI menggunakan halaman HTML jika diperlukan. Namun, dalam modul ini kita akan melihat pembuatan aplikasi web di R.

Aplikasi Shiny

- Aplikasi Shiny terdapat dalam satu skrip yang disebut **app.R**.
- Skrip app.R berada di suatu direktori, misalnya direktori **newdir/** dan aplikasi dapat dijalankan dengan **runApp("newdir")**.
- Skrip app.R terdiri dari 3 komponen, yaitu:
 - a. **objek antarmuka pengguna (UI)**, berfungsi untuk mengontrol tata letak dan tampilan aplikasi yang akan dilihat pengguna di aplikasi web serta menjabarkan input atau masukan dari pengguna.
 - b. **fungsi server**, berisi instruksi yang dibutuhkan komputer untuk membangun aplikasi, misalnya logika aplikasi yang menghasilkan plot.
 - c. **panggilan ke fungsi shinyApp**, fungsi shinyApp membuat objek aplikasi Shiny dari pasangan UI/server eksplisit.

Aplikasi Shiny dari Contoh Bawaan Paket shiny

✗ Jika belum menginstal paket Shiny

buka sesi R, sambungkan ke internet, dan jalankan

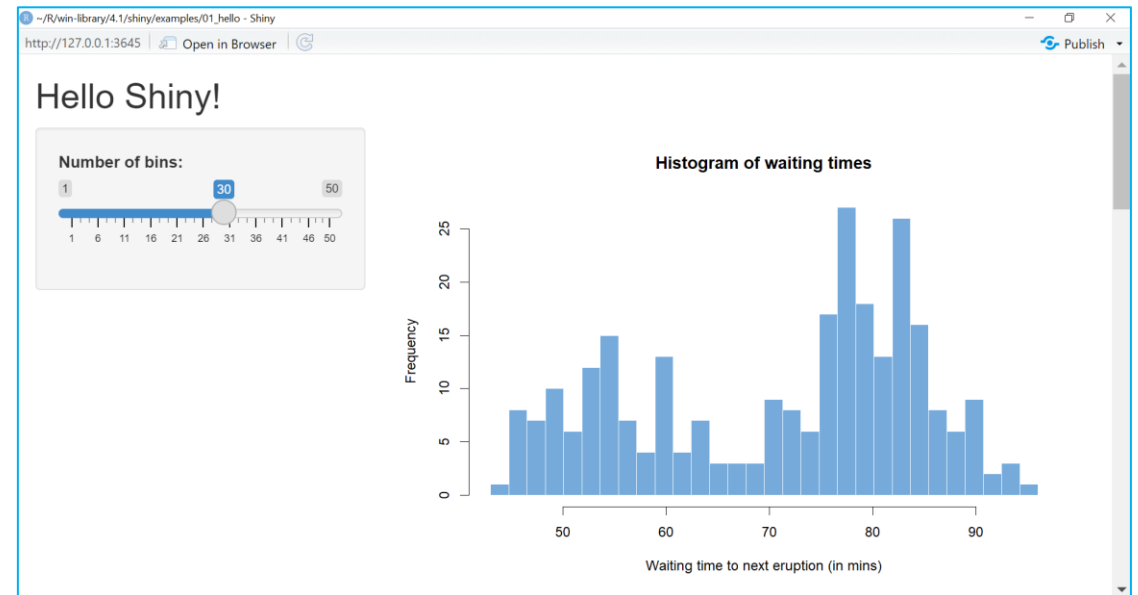
```
install.packages("shiny")
```

Paket shiny memiliki 11 contoh bawaan yang masing-masing menunjukkan cara kerja Shiny.

Contoh 1 : Hello Shiny

Membuat plot histogram dari dataset R dengan jumlah bin yang dapat dikonfigurasi dengan cara menggeser bilah penggeser, lalu aplikasi akan merespon masukan dari pengguna.

```
library(shiny)  
runExample("01_hello")
```



Gambar 1

Skrip app.R dari Hello Shiny

```
library(shiny)

# Define UI for app that draws a histogram ----
ui <- fluidPage(

  # App title ----
  titlePanel("Hello Shiny!"),

  # Sidebar layout with input and output definitions ----
  sidebarLayout(

    # Sidebar panel for inputs ----
    sidebarPanel(

      # Input: Slider for the number of bins ----
      sliderInput(inputId = "bins",
                  label = "Number of bins:",
                  min = 1,
                  max = 50,
                  value = 30)

    ),

    # Main panel for displaying outputs ----
    mainPanel(

      # Output: Histogram ----
      plotOutput(outputId = "distPlot")

    )

  )
)
```

UI

```
# Define server logic required to draw a histogram ----
server <- function(input, output) {

  # Histogram of the Old Faithful Geyser Data ----
  # with requested number of bins
  # This expression that generates a histogram is wrapped in a call
  # to renderPlot to indicate that:
  #
  # 1. It is "reactive" and therefore should be automatically
  #    re-executed when inputs (input$bins) change
  # 2. Its output type is a plot
  output$distPlot <- renderPlot({

    x <- faithful$waiting
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

    hist(x, breaks = bins, col = "#75AADB", border = "white",
         xlab = "Waiting time to next eruption (in mins)",
         main = "Histogram of waiting times")

  })

}

# Create Shiny app ----
shinyApp(ui = ui, server = server)
```

Fungsi
Server

Fungsi shinyApp

Kode sumber ini terdiri dari 3 komponen :

1. UI
2. fungsi server
3. panggilan ke fungsi shinyApp

Contoh 1 : Hello Shiny

- **Bagian UI** memuat `fluidPage()` untuk mengontrol tata letak dan tampilan dari aplikasi. Fungsi `slidebar()` untuk menampilkan batang penggeser sebagai input bagi pengguna untuk mengatur sendiri banyaknya bin dari histogram sehingga visualisasi bersifat interaktif.
- Pada **fungsi server** Hello Shiny, script melakukan beberapa perhitungan dan kemudian memplot histogram sesuai dengan jumlah bin yang diminta pengguna (diatur di bagian UI). Sebagian besar skrip fungsi server dibungkus dengan panggilan ke `renderPlot`. Komentar di atas fungsi menjelaskan sedikit tentang ini.
- **Fungsi `shinyApp()`** untuk memanggil pasangan UI dan server agar dihasilkan aplikasi Shiny.

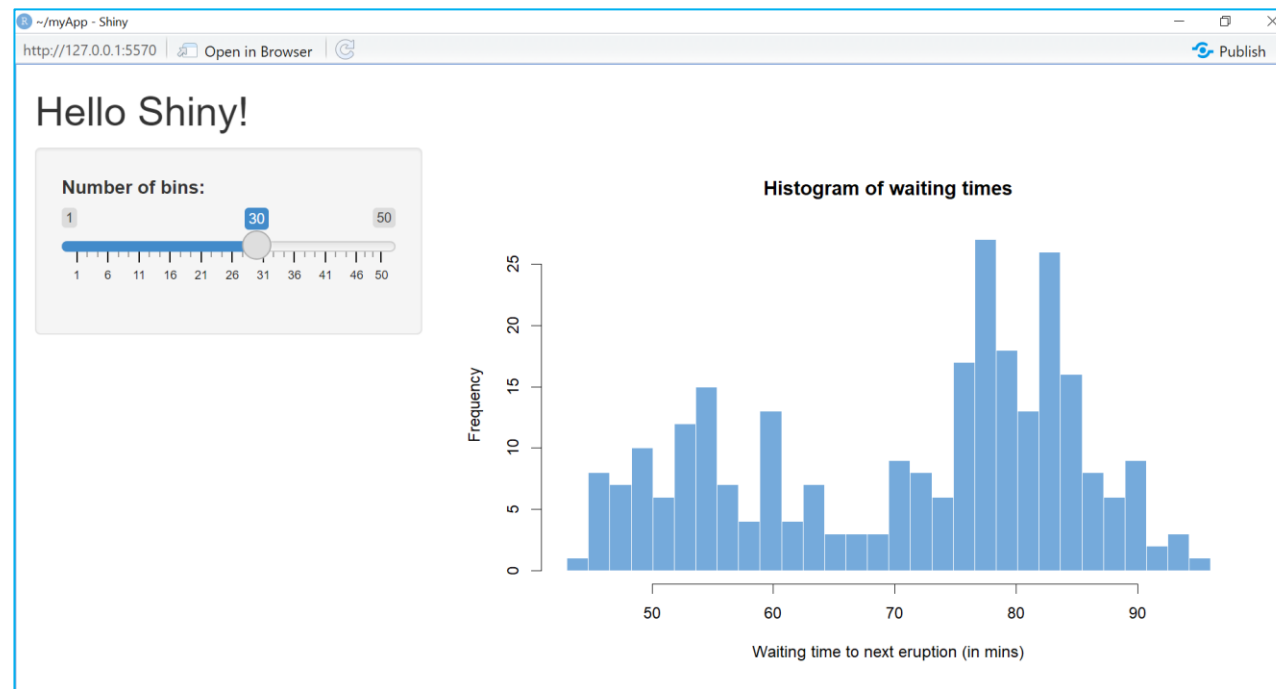
Setelah semua instruksi dijalankan di RGui/ RStudio, aplikasi web muncul seperti dalam Gambar 1.



- Saat **aplikasi aktif**, sesi **R akan sibuk** sehingga tidak dapat menjalankan perintah R apa pun.
- Untuk mengaktifkan kembali sesi R, tekan **escape** atau klik ikon **STOP**.

- Aplikasi Shiny juga dapat dibangun dengan **membuat direktori baru** dan menyimpan file app.R di dalamnya.
- Aplikasi Shiny dapat dijalankan dengan **memberikan nama direktorinya** ke fungsi **runApp()**.
- Misalkan kode aplikasi Hello Shiny disimpan di direktori **~/myApp** atau **myApp/app.R**

```
library(shiny)
runApp("~/myApp") #atau runApp("myApp/App.R")
```



Gambar 2

Contoh lainnya:

```
runExample("02_text")      # tables and data frames
runExample("03_reactivity") # a reactive expression
runExample("04_mpg")       # global variables
runExample("05_sliders")   # slider bars
runExample("06_tabsets")   # tabbed panels
runExample("07_widgets")   # help text and submit buttons
runExample("08_html")      # Shiny app built from HTML
runExample("09_upload")    # file upload wizard
runExample("10_download")  # file download wizard
runExample("11_timer")     # an automated timer
```

- Jika semua kode diatas dijalankan secara langsung, aplikasi contoh Shiny terbuka satu per satu sekaligus dengan app.R di dalam layar.
- Aplikasi yang pertama kali muncul adalah Shiny Text, lalu setelah ditutup, muncul aplikasi Reactivity, begitu seterusnya hingga contoh aplikasi ke-11.

Contoh 2

```
runExample("02_text")
```

tables and data frames

Shiny Text

Choose a dataset:

rock

Number of observations to view:

10

area	peri	shape	perm
Min. : 1016	Min. : 308.6	Min. : 0.09033	Min. : 6.30
1st Qu.: 5305	1st Qu.: 1414.9	1st Qu.: 0.16226	1st Qu.: 76.45
Median : 7487	Median : 2536.2	Median : 0.19886	Median : 130.50
Mean : 7188	Mean : 2682.2	Mean : 0.21811	Mean : 415.45
3rd Qu.: 8870	3rd Qu.: 3989.5	3rd Qu.: 0.26267	3rd Qu.: 777.50
Max. : 12212	Max. : 4864.2	Max. : 0.46413	Max. : 1300.00

area	peri	shape	perm
4990	2791.90	0.09	6.30
7002	3892.60	0.15	6.30
7558	3930.66	0.18	6.30
7352	3869.32	0.12	6.30
7943	3948.54	0.12	17.10
7979	4010.15	0.17	17.10
9333	4345.75	0.19	17.10
8209	4344.75	0.16	17.10
8393	3682.04	0.20	119.00
6425	3098.65	0.16	119.00

Gambar 3

Contoh 3

```
runExample("03_reactivity") # a reactive expression
```

Reactivity

Caption:

Choose a dataset:

Number of observations to view:

Data Summary

area	peri	shape	perm
Min. : 1016	Min. : 308.6	Min. : 0.09033	Min. : 6.30
1st Qu.: 5305	1st Qu.: 1414.9	1st Qu.: 0.16226	1st Qu.: 76.45
Median : 7487	Median : 2536.2	Median : 0.19886	Median : 130.50
Mean : 7188	Mean : 2682.2	Mean : 0.21811	Mean : 415.45
3rd Qu.: 8870	3rd Qu.: 3989.5	3rd Qu.: 0.26267	3rd Qu.: 777.50
Max. : 12212	Max. : 4864.2	Max. : 0.46413	Max. : 1300.00

area	peri	shape	perm
4990	2791.90	0.09	6.30
7002	3892.60	0.15	6.30
7558	3930.66	0.18	6.30
7352	3869.32	0.12	6.30
7943	3948.54	0.12	17.10
7979	4010.15	0.17	17.10
9333	4345.75	0.19	17.10
8209	4344.75	0.16	17.10
8393	3682.04	0.20	119.00
6425	3098.65	0.16	119.00

Gambar 4

Contoh 4

```
runExample("04_mpg")
```

```
# global variables
```

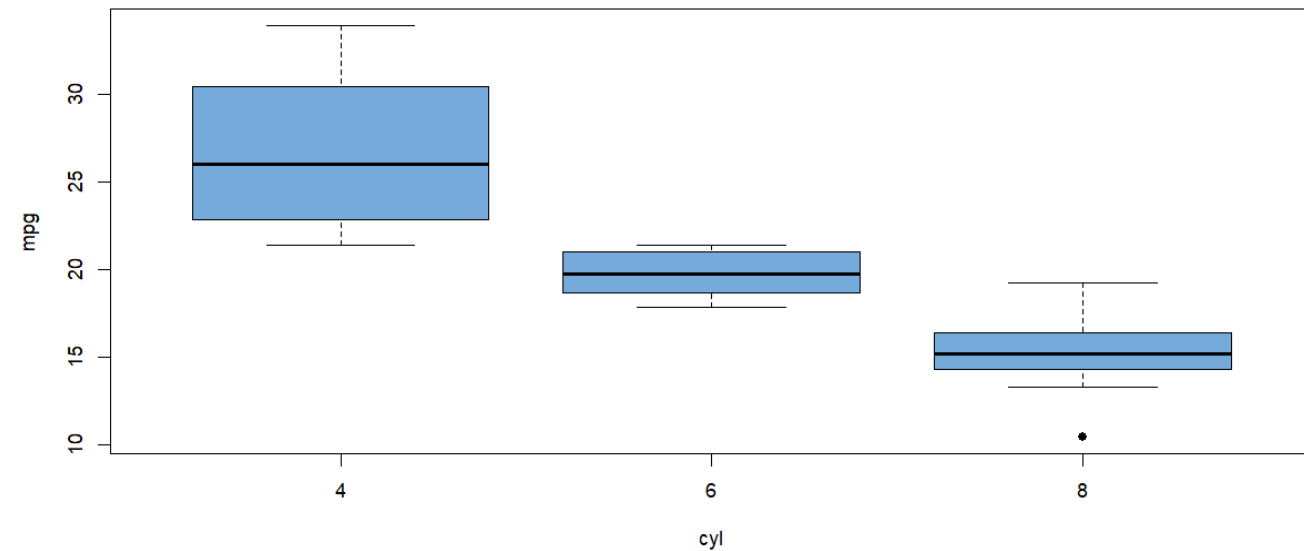
Miles Per Gallon

Variable:

Cylinders

☒ Show outliers

mpg ~ cyl

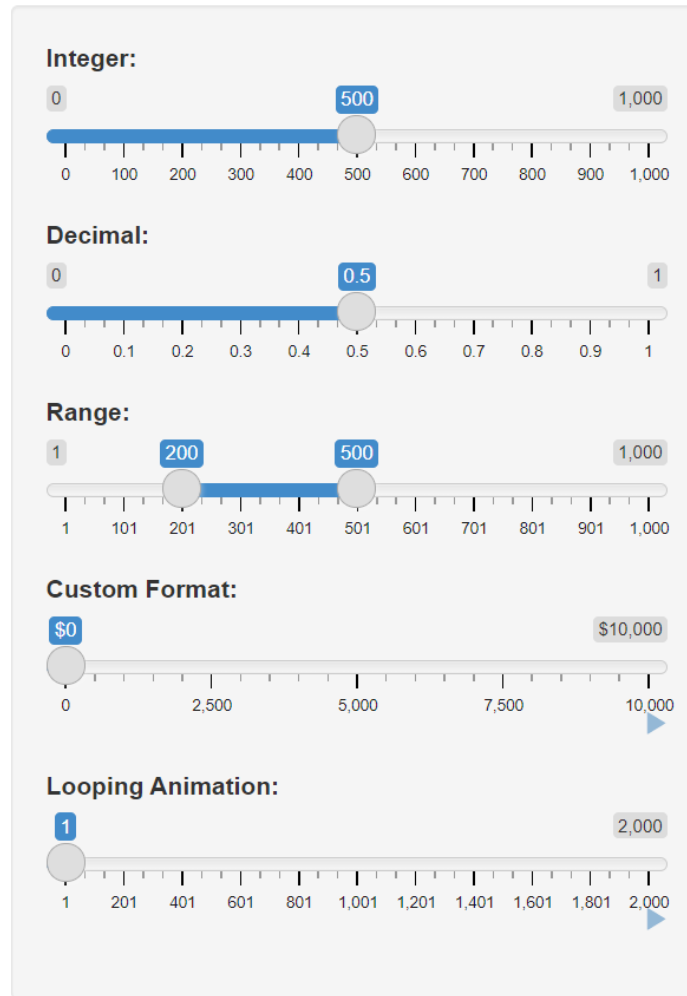


Gambar 5

Contoh 5

```
runExample("05_sliders") # slider bars
```

Sliders



Name	Value
Integer	500
Decimal	0.5
Range	200 500
Custom Format	0
Animation	1

Gambar 6

Contoh 6

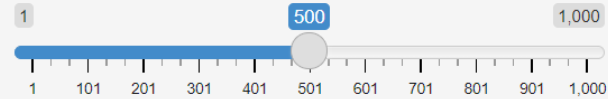
```
runExample("06_tabsets") # tabbed panels
```

Tabsets

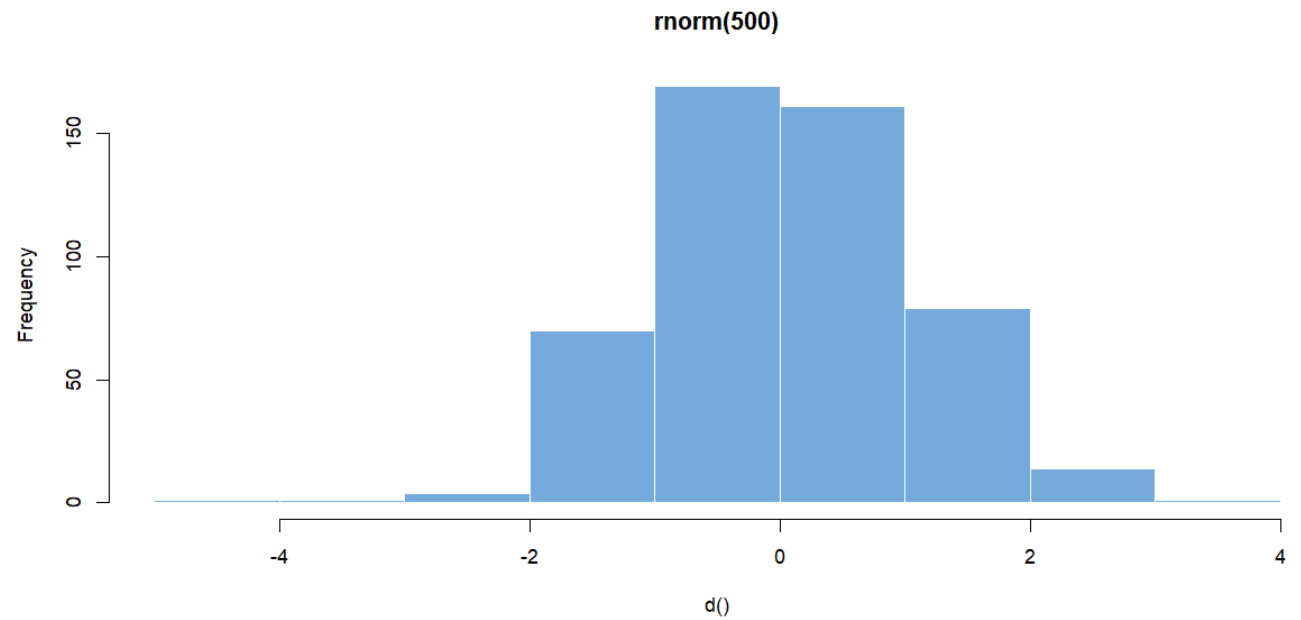
Distribution type:

- ☒ Normal
- ☐ Uniform
- ☐ Log-normal
- ☐ Exponential

Number of observations:



Plot Summary Table



Gambar 7

Contoh 7

```
runExample("07_widgets") # help text and submit buttons
```

More Widgets

Choose a dataset:

rock ▼

Number of observations to view:

10

Note: while the data view will show only the specified number of observations, the summary will still be based on the full dataset.

Update View

Summary

area	peri	shape	perm
Min. : 1016	Min. : 308.6	Min. : 0.09033	Min. : 6.30
1st Qu.: 5305	1st Qu.: 1414.9	1st Qu.: 0.16226	1st Qu.: 76.45
Median : 7487	Median : 2536.2	Median : 0.19886	Median : 130.50
Mean : 7188	Mean : 2682.2	Mean : 0.21811	Mean : 415.45
3rd Qu.: 8870	3rd Qu.: 3989.5	3rd Qu.: 0.26267	3rd Qu.: 777.50
Max. : 12212	Max. : 4864.2	Max. : 0.46413	Max. : 1300.00

Observations

area	peri	shape	perm
4990	2791.90	0.09	6.30
7002	3892.60	0.15	6.30
7558	3930.66	0.18	6.30
7352	3869.32	0.12	6.30
7943	3948.54	0.12	17.10
7979	4010.15	0.17	17.10
9333	4345.75	0.19	17.10
8209	4344.75	0.16	17.10
8393	3682.04	0.20	119.00
6425	3098.65	0.16	119.00

Gambar 8

Contoh 8

```
runExample("08_html")
```

```
# Shiny app built from HTML
```

HTML UI

Distribution type:

Normal

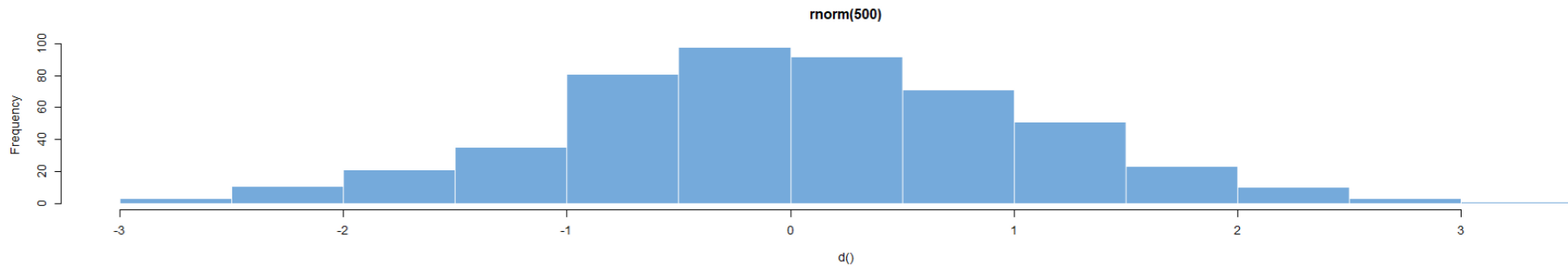
Number of observations:

500

Summary of data:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-2.903607	-0.638970	0.006116	0.023787	0.706622	3.018497

Plot of data:



Head of data:

x

-0.16
-0.69
0.45
-0.52
-0.10
-2.08

Gambar 9

Contoh 9

```
runExample("09_upload")    # file upload wizard
```

Uploading Files

Choose CSV File

Browse...

No file selected

☒ Header

Separator

☒ Comma

☐ Semicolon

☐ Tab

Quote

☐ None

☒ Double Quote

☐ Single Quote

Display

☒ Head

☐ All

Gambar 10

Contoh 10

```
runExample("10_download") # file download wizard
```

Downloading Data

Choose a dataset:

rock ▼

Download

area	peri	shape	perm
4990	2791.90	0.09	6.30
7002	3892.60	0.15	6.30
7558	3930.66	0.18	6.30
7352	3869.32	0.12	6.30
7943	3948.54	0.12	17.10
7979	4010.15	0.17	17.10
9333	4345.75	0.19	17.10
8209	4344.75	0.16	17.10
8393	3682.04	0.20	119.00
6425	3098.65	0.16	119.00
9364	4480.05	0.15	119.00
8624	3986.24	0.15	119.00
10651	4036.54	0.23	82.40
8868	3518.04	0.23	82.40
9417	3999.37	0.17	82.40
8874	3629.07	0.15	82.40
10962	4608.66	0.20	58.60
10743	4787.62	0.26	58.60

Gambar 11

Contoh 11

```
runExample("11_timer")      # an automated timer
```

The current time is 2021-09-04 11:35:22

The function `invalidateLater()` can be used to invalidate an observer or reactive expression in a given number of milliseconds. In this example, the output `currentTime` is updated every second, so it shows the current time on a second basis.

app.R

↑ show with app

```
library(shiny)

# Define UI for displaying current time ----
ui <- fluidPage(

  h2(textOutput("currentTime"))

)

# Define server logic to show current time, update every second ----
server <- function(input, output, session) {

  output$currentTime <- renderText({
    invalidateLater(1000, session)
    paste("The current time is", Sys.time())
  })

}

# Create Shiny app ----
shinyApp(ui, server)
```

Gambar 12

Membuat Aplikasi Shiny

```
library(shiny)

# Define UI ----
ui <- fluidPage(

)

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

}

# Run the app ----
shinyApp(ui = ui, server = server)
```



HALAMAN KOSONG

```
library(shiny)

# Define UI ----
ui <- fluidPage(

)

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

}

# Run the app ----
shinyApp(ui = ui, server = server)
```



title panel

sidebar panel

main panel

Gambar 13

Widget

Aplikasi Shiny dapat menampilkan widget untuk **menambah tampilan** yang ada di halaman web atau antarmuka pengguna.

Tabel 1 Fungsi untuk menampilkan widget

No	Fungsi	Widget
1	actionBotton	Tombol aksi
2	checkboxGroupInput	Sekelompok kotak centang
3	checkboxInput	Satu kotak centang
4	dateInput	Kalender untuk membantu pemilihan tanggal
5	dateRangeInput	Sepasang kalender untuk memilih rentang tanggal
6	fileInput	Control wizard unggah file
7	helpText	Teks bantuan yang dapat ditambahkan ke formulir input
8	numericInput	Bidang untuk memasukkan angka
9	radioButtons	Satu set tombol radio
10	selectInput	Sebuah kotak dengan pilihan untuk dipilih
11	sliderInput	Bilah penggeser
12	submitButton	Tombol kirim
13	textInput	Bidang untuk memasukkan teks

```
library(shiny)
```

```
# Define UI ----
```

```
ui <- fluidPage(
  titlePanel("Basic widgets"),

  fluidRow(

    column(3,
      h3("Buttons"),
      actionButton("action", "Action"),
      br(),
      br(),
      submitButton("Submit")),
    column(3,
      h3("Single checkbox"),
      checkboxInput("checkbox", "Choice A", value = TRUE)),

    column(3,
      checkboxGroupInput("checkGroup",
        h3("Checkbox group"),
        choices = list("Choice 1" = 1,
          "Choice 2" = 2,
          "Choice 3" = 3),
        selected = 1)),

    column(3,
      dateInput("date",
        h3("Date input"),
        value = "2014-01-01"))
  ),

  fluidRow(

    column(3,
      dateRangeInput("dates", h3("Date range"))),
    column(3,
      fileInput("file", h3("File input"))),
    column(3,
      h3("Help text"),
      helpText("Note: help text isn't a true widget,",
        "but it provides an easy way to add text to",
        "accompany other widgets."))
    column(3,
      numericInput("num",
        h3("Numeric input"),
        value = 1))
  )
)
```

Widget

```
fluidRow(

  column(3,
    radioButtons("radio", h3("Radio buttons"),
      choices = list("Choice 1" = 1, "Choice 2" = 2,
        "Choice 3" = 3), selected = 1)),

  column(3,
    selectInput("select", h3("Select box"),
      choices = list("Choice 1" = 1, "Choice 2" = 2,
        "Choice 3" = 3), selected = 1)),

  column(3,
    sliderInput("slider1", h3("Sliders"),
      min = 0, max = 100, value = 50),
    sliderInput("slider2", "",
      min = 0, max = 100, value = c(25, 75))
  ),

  column(3,
    textInput("text", h3("Text input"),
      value = "Enter text..."))
)

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

}

# Run the app ----
shinyApp(ui = ui, server = server)
```

Widget

OUTPUT :

Basic widgets

Buttons <input type="button" value="Action"/> <input type="button" value="Submit"/>	Single checkbox <input checked="" type="checkbox"/> Choice A	Checkbox group <input checked="" type="checkbox"/> Choice 1 <input type="checkbox"/> Choice 2 <input type="checkbox"/> Choice 3	Date input <input type="text" value="2014-01-01"/>
Date range <input type="text" value="2021-09-04"/> to <input type="text" value="2021-09-04"/>	File input <input type="button" value="Browse..."/> <input type="text" value="No file selected"/>	Help text <small>Note: help text isn't a true widget, but it provides an easy way to add text to accompany other widgets.</small>	Numeric input <input type="text" value="1"/>
Radio buttons <input checked="" type="radio"/> Choice 1 <input type="radio"/> Choice 2 <input type="radio"/> Choice 3	Select box <input type="text" value="Choice 1"/>	Sliders <input type="range" value="50"/> <input type="range" value="25"/>	Text input <input type="text" value="Enter text..."/>

Gambar 14

Output Reaktif

Tabel 2 Fungsi Output

Fungsi Output	Membuat
<code>dataTableOutput</code>	Tabel Data
<code>htmlOutput</code>	HTML mentah
<code>imageOutput</code>	Gambar
<code>plotOutput</code>	Plot
<code>tableOutput</code>	Table
<code>textOutput</code>	Text
<code>uiOutput</code>	HTML mentah

Tabel 3 Fungsi render

Fungsi Output	Membuat
<code>renderDataTable</code>	Tabel data
<code>renderImage</code>	gambar (disimpan sebagai tautan ke file sumber)
<code>renderPlot</code>	Plot
<code>renderPrint</code>	hasil cetak apa pun
<code>renderTable</code>	bingkai data, matriks, tabel lain seperti struktur
<code>renderText</code>	string karakter
<code>renderUI</code>	objek tag Shiny atau HTML


```
library(shiny)
```

```
# Define UI ----
```

```
ui <- fluidPage(
  titlePanel("censusVis"),

  sidebarLayout(
    sidebarPanel(
      helpText("Create demographic maps with
        information from the 2010 US Census."),

      selectInput("var",
        label = "Choose a variable to display",
        choices = c("Percent White",
                     "Percent Black",
                     "Percent Hispanic",
                     "Percent Asian"),
        selected = "Percent White"),

      sliderInput("range",
        label = "Range of interest:",
        min = 0, max = 100, value = c(0, 100))
    ),

    mainPanel(
      textOutput("selected_var")
    )
  )
)
```

```
# Define server logic ----
```

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

  output$selected_var <- renderText({
    "You have selected this"
  })

}
```

```
# Run the app ----
```

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

Output Reaktif

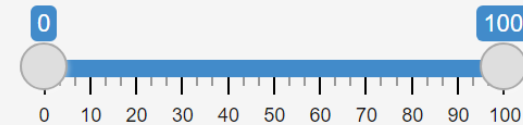
censusVis

Create demographic maps with
information from the 2010 US
Census.

Choose a variable to display

Percent White ▼

Range of interest:



You have selected this

Gambar 15

Output Reaktif

```
library(shiny)

# Define UI ----
ui <- fluidPage(
  titlePanel("censusVis"),

  sidebarLayout(
    sidebarPanel(
      helpText("Create demographic maps with
        information from the 2010 US Census."),

      selectInput("var",
        label = "Choose a variable to display",
        choices = c("Percent White",
                    "Percent Black",
                    "Percent Hispanic",
                    "Percent Asian"),
        selected = "Percent White"),

      sliderInput("range",
        label = "Range of interest:",
        min = 0, max = 100, value = c(0, 100))
    ),

    mainPanel(
      textOutput("selected_var")
    )
  )

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

  output$selected_var <- renderText({
    paste("You have selected this", input$var)
  })

}

# Run the app ----
shinyApp(ui = ui, server = server)
```

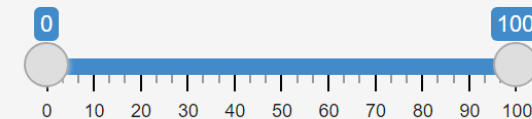
censusVis

Create demographic maps with
information from the 2010 US Census.

Choose a variable to display

Percent White ▼

Range of interest:



You have selected Percent White

Gambar 16

Kode aplikasi Shiny disimpan lalu ditampilkan dengan mode showcase

```
runApp('censusVis/census.R', display.mode = "showcase")
```

Membuat Dashboard

```
install.packages("shinydashboard")
```

```
library(shiny)
library(shinydashboard)

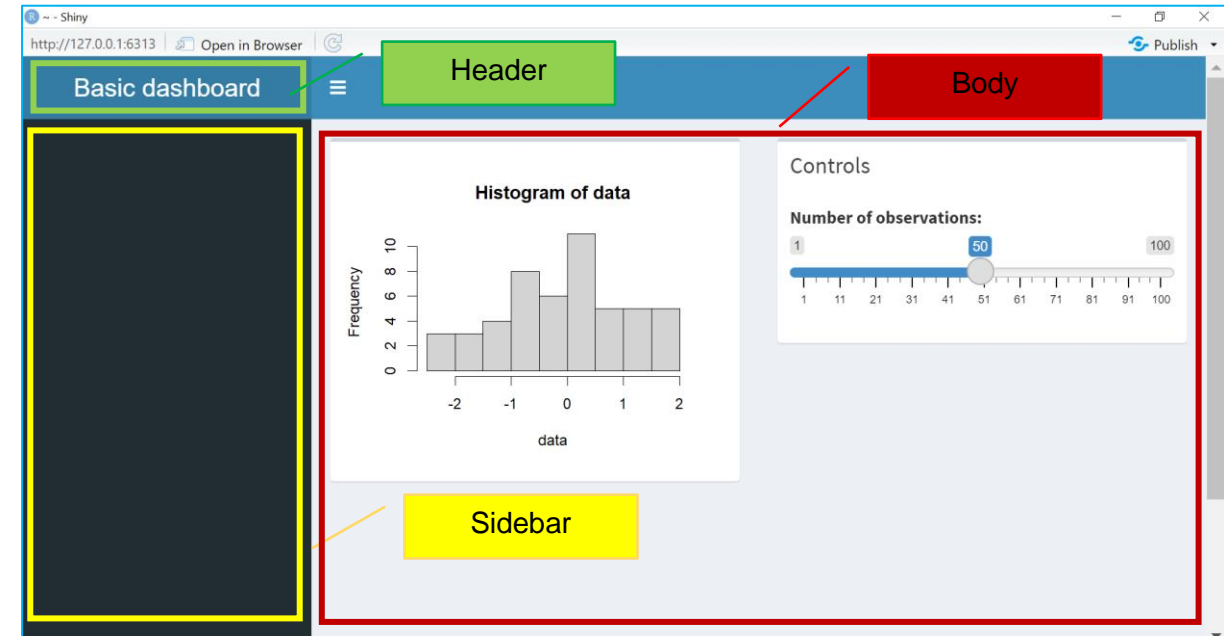
ui <- dashboardPage(
  dashboardHeader(title = "Basic dashboard"),
  dashboardSidebar(),
  dashboardBody(
    # Boxes need to be put in a row (or column)
    fluidRow(
      box(plotOutput("plot1", height = 250)),

      box(
        title = "Controls",
        sliderInput("slider", "Number of observations:", 1, 100, 50)
      )
    )
  )

server <- function(input, output) {
  set.seed(122)
  histdata <- rnorm(500)

  output$plot1 <- renderPlot({
    data <- histdata[seq_len(input$slider)]
    hist(data)
  })
}

shinyApp(ui, server)
```



Gambar 17

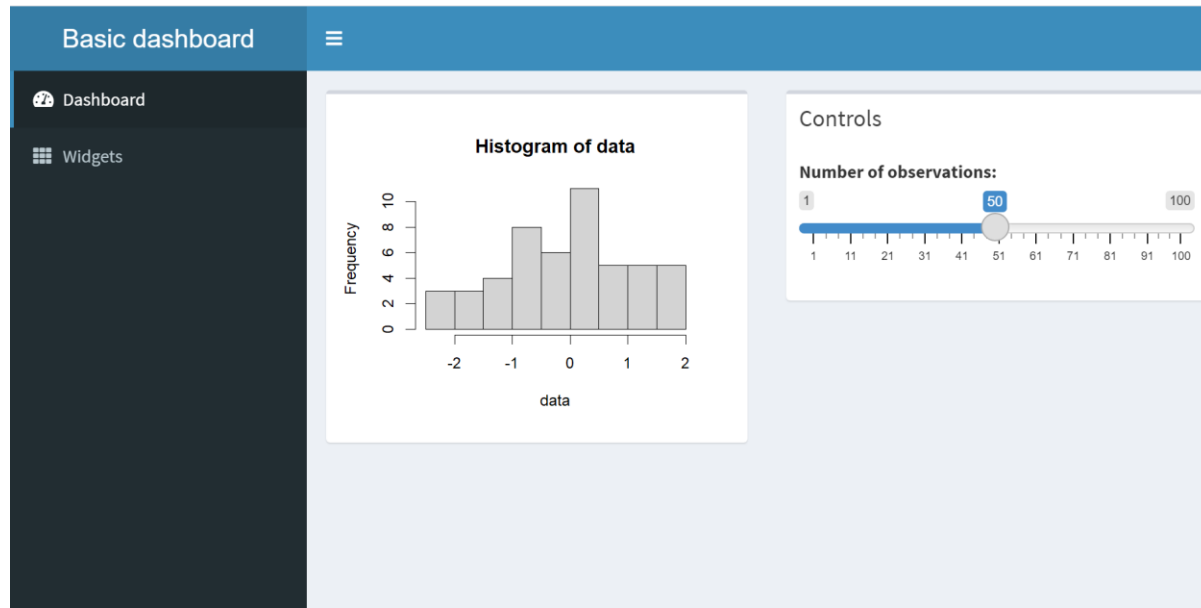
Membuat Dashboard

```
ui <- dashboardPage(  
  dashboardHeader(title = "Basic dashboard"),  
  ## Sidebar content  
  dashboardSidebar(  
    sidebarMenu(  
      menuItem("Dashboard", tabName = "dashboard", icon = icon("dashboard")),  
      menuItem("Widgets", tabName = "widgets", icon = icon("th"))  
    )  
  ),  
  ## Body content  
  dashboardBody(  
    tabItems(  
      # First tab content  
      tabItem(tabName = "dashboard",  
        fluidRow(  
          box(plotOutput("plot1", height = 250)),  
  
          box(  
            title = "Controls",  
            sliderInput("slider", "Number of observations:", 1, 100, 50)  
          )  
        )  
      ),  
      # Second tab content  
      tabItem(tabName = "widgets",  
        h2("Widgets tab content")  
      )  
    )  
  )  
)
```

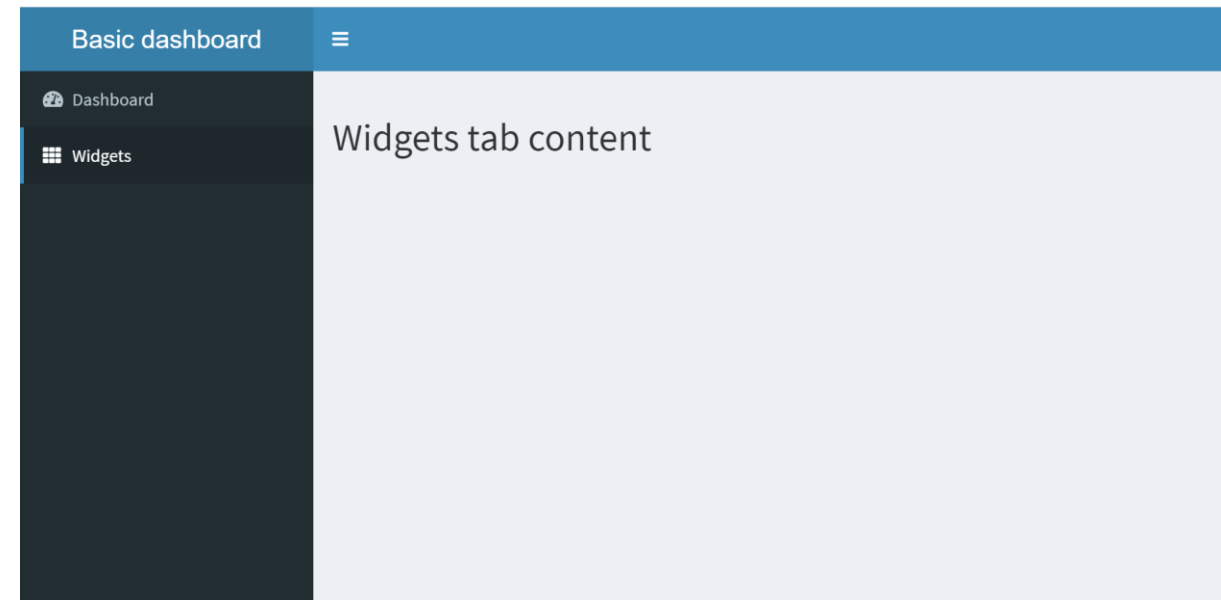
Jika tidak ingin menampilkan sidebar:

```
dashboardSidebar(disable = TRUE)
```

OUTPUT



(a)



(b)

Gambar 18

Percobaan Praktikum & Tugas



Referensi

Pathak, M. A., (2014), *Beginning Data Science with R*, Springer International Publishing, Switzerland.

Dietrich, D., Barry H., & Beibei Y., (2015), *Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*, John Wiley & Sons, Inc., Indianapolis, Indiana

<https://shiny.rstudio.com/tutorial/>

http://homepage.stat.uiowa.edu/~rdecook/stat6220/Class_notes/R-shiny_intro.pdf#:~:text=To%20make%20an%20R%20Shiny%20app%2C%20start%20with,into%20a%20single%20folder%20named%20for%20your%20app

<https://shiny.rstudio.com/tutorial/written-tutorial/lesson1/>

http://rstudio.github.io/shinydashboard/get_started.html

<https://shiny.rstudio.com/articles/plot-interaction.html>



KMMI 2021

Eksplorasi dan Visualisasi Data

Pertemuan 10:
Visualisasi Menggunakan R Shiny