(原始code)

# This is the user-interface definition of a Shiny web application.

# You can find out more about building applications with Shiny here:

#

# http://shiny.rstudio.com

#

library(shiny)

shinyUI(fluidPage(

# Application title

titlePanel("Old Faithful Geyser Data"),

# Sidebar with a slider input for number of bins

sidebarLayout(

sidebarPanel(

sliderInput("bins",

"Number of bins:",

min = 1,

max = 50,

value = 30)

),

# Show a plot of the generated distribution

mainPanel(

plotOutput("distPlot")

)

)

))

(藍色部分是老師加的)

# This is the user-interface definition of a Shiny web application.

# You can find out more about building applications with Shiny here:

#

# http://shiny.rstudio.com

#

library(shiny)

shinyUI(fluidPage(

# Application title

titlePanel("Old Faithful Geyser Data"),

# Sidebar with a slider input for number of bins

sidebarLayout(

sidebarPanel(

sliderInput("bins",

"Number of bins:",

min = 1,

max = 50,

value = 30),

numericInput("inputn",

"input number",

value = 20)

),

# Show a plot of the generated distribution

mainPanel(

plotOutput("distPlot"),

plotOutput("testPlot")

)

)

))

# This is the server logic for a Shiny web application.

# You can find out more about building applications with Shiny here:

#

# http://shiny.rstudio.com

#

library(shiny)

shinyServer(function(input, output) {

output$distPlot <- renderPlot({

# generate bins based on input$bins from ui.R

x <- faithful[, 2]

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

# draw the histogram with the specified number of bins

hist(x, breaks = bins, col = 'darkgray', border = 'white')

})

})

# This is the server logic for a Shiny web application.

# You can find out more about building applications with Shiny here:

#

# http://shiny.rstudio.com

#

library(shiny)

shinyServer(function(input, output) {

output$distPlot <- renderPlot({

# generate bins based on input$bins from ui.R

x <- faithful[, 2]

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

# draw the histogram with the specified number of bins

hist(x, breaks = bins, col = 'darkgray', border = 'white')

})

output$testPlot <- renderPlot({

getUIinput = input$inputn

plot(1:getUIinput)

})

})