

```

library(shiny)
library(datasets)

# Define UI for application that draws a histogram
shinyUI(fluidPage(

  # Application title
  titlePanel("Number of telephones by region and year"),

  # Sidebar with a slider input for the number of bins
  sidebarLayout(
    sidebarPanel(
      checkboxInput("region_check", "Region", TRUE),
      checkboxInput("year_check", "Year", FALSE),
      selectInput("region", label = h3("Region:"),
                  choices = colnames(WorldPhones), selected = 1),

      selectInput("year", label = h3("Year:"),
                  choices = rownames(WorldPhones), selected = 1)
    ),

    # Show a plot of the generated distribution
    mainPanel(
      plotOutput("barPlot")
    )
  )
))

```

```

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

  # Expression that generates a histogram. The expression is
  # wrapped in a call to renderPlot to indicate that:
  #
  # 1) It is "reactive" and therefore should re-execute automatically
  #    when inputs change
  # 2) Its output type is a plot

  output$barPlot <- renderPlot({
    # draw the bargraph based on x-axis selection
    #select data by partitioning dataset
    if (input$region_check) {dataset<- WorldPhones[,input$region]}
    else if (input$year_check) {dataset<- WorldPhones[input$year,]}
    else {dataset<- WorldPhones[,input$region]}

    #set title and axis labels
    if (input$region_check) {ttl<- input$region}
    else if (input$year_check) {ttl<- input$year}
    else {ttl<- input$region}

    if (input$region_check) {x_axis<- 'Year'}
    else if (input$year_check) {x_axis<- 'Region'}
    else {x_axis<- "Year"}

    barplot(dataset, main = ttl, xlab = x_axis, ylab = 'Number of Telephones', col ="blue")
  })
})

```

Number of telephones by region and year

☒ Region

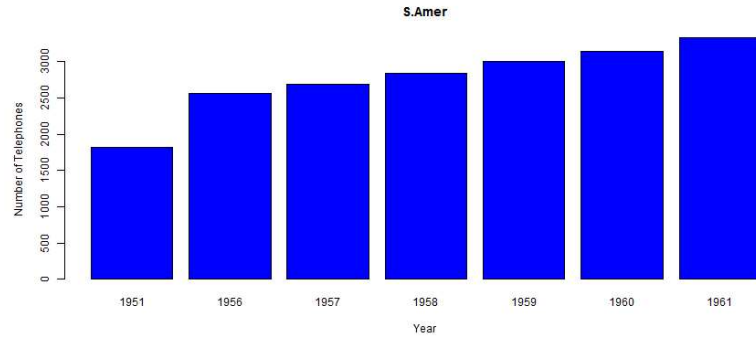
☐ Year

Region:

S Amer

Year:

1956



Number of telephones by region and year

☐ Region

☒ Year

Region:

S Amer

Year:

1956

