

Shiny app for beginners

Activity #1: Start the demo app

1. Start the demo app. If you are using RStudio from your own laptop, go to **New project → New Directory → Shiny app.** If you are on Posit.cloud, go to **New Project → New RStudio project → File → New File → Shiny app.**
2. Name the app **Shiny_demo**. This will be come a name of a folder.
3. Run the app.

Activity #2: Making small changes

1. Change the application title to “Shiny demo”. Run the app to see the effect.

```
15      # Application title  
16      titlePanel("Shiny Demo"),
```

2. Change the color from “darkgray” to “green”. Run the app.

```
44      hist(x, breaks = bins, col = 'green', border = 'white',  
45          xlab = 'Waiting time to next eruption (in mins)',  
46          main = 'Histogram of waiting times')
```

3. Swap data to mtcars

```
40      x     <- mtcars[, 'mpg']
```

Activity #3: Adding and using control widgets

1. Bring the Shiny Cheat sheet: Help → Cheat Sheets → Web application with Shiny
2. Add a color select to the UI. Add these 5 lines. Run the app. Make sure there is no errors and it looks good.

```
19      sidebarLayout(  
20          sidebarPanel(  
21              |      selectInput("select_color",  
22                  "Select a color",  
23                  choices = c("red", "green", "gray")  
24              ),  
25              sliderInput("bins",  
26                  "Number of bins:",  
27                  min = 1,
```

3. Change the color of the plot. Change this line and run the app.

```
47      # draw the histogram with the specified number of bins  
48      hist(x, breaks = bins, col = input$select_color, border = 'white',  
49          xlab = 'Waiting time to next eruption (in mins)',
```

Activity #4: Change data columns

1. Add a selectInput to select the columns. Run the app to make sure the control widget works.

```

18     # sidebar with a slider input for number of bins
19     sidebarLayout(
20         sidebarPanel(
21             selectInput("select_column",
22                         "Select a column",
23                         choices = colnames(mtcars)
24             ),
25             selectInput("select_color",
26                         "Select a color",
27                         choices = c("red", "green", "gray"))

```

2. Use the select column to plot. Run the app.

```

47     # generate bins based on input$bins from ui.R
48     x      <- mtcars[, input$select_column]
49     bins <- seq(min(x), max(x), length.out = input$bins + 1)

```

Activity 5. Add a plot.

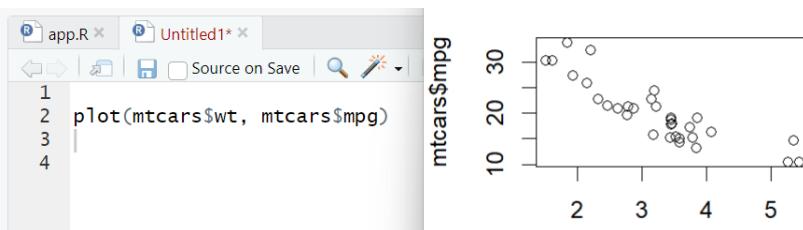
1. First add the plotoutput in UI. Run the app.

```

37     mainPanel(
38         plotOutput("distPlot"),
39         plotOutput("sPlot")
40     )
41 )
42 )

```

2. Generate the plot in a separate R script.



3. Copy the code into the Shiny app inside a output function.

```

44 # Define server logic required to draw a histogram
45 server <- function(input, output) {
46
47   |   output$sPlot <- renderPlot({
48     plot(mtcars$wt, mtcars$mpg)
49   })
50
51   output$distPlot <- renderPlot({
52     # generate bins based on input$bins from ui.R

```

4. Change the code to use selected columns.

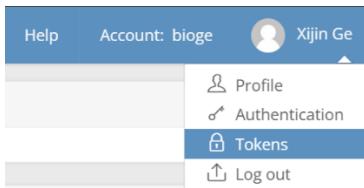
```

47 ▾   output$sPlot <- renderPlot({
48       plot(mtcars[, input$select_column], mtcars$mpg)
49 ▾ })

```

Activity 6: Publishing the app on Shinyapp.io

1. Log into Shinyapps.io
2. Click on the top right corner (your name). Then select Tokens.



3. Click on **+ Add Token**
4. Click on Show Token and the **Show Secret**. Copy to clipboard.

To set up the `rsconnect` package, click the copy button below and paste the command into the R console.

```
rsconnect::setAccountInfo(name='bioge',
                          token='46EBE9B988F6525C47F96A9963ECF216',
                          secret='VfCFelvTeXt4Xx54tbbxX1BstBkbiOc6RmfftAL1')
```

Hide secret

Copy to clipboard

www.shinyapps.io says

Copy to clipboard **Ctrl+C** Enter

`'96A9963ECF216', secret='VfCFelvTeXt4Xx54tbbxX1BstBkbiOc6RmfftAL1')`

OK

Cancel

5. Go back to RStudio. Run the app. Click the Publish button on the top right.

The screenshot shows the RStudio interface with a Shiny application running at `http://127.0.0.1:7666`. The application has a title 'Shiny Demo' and a sidebar with 'Select a column' (set to 'mpg') and 'Select a color'. To the right is a histogram titled 'Histogram of waiting times' with a single red bar. At the top right of the RStudio window, there is a 'Publish' button, which is circled in red.

6. Connect to Shinyapps.io by pasting the secret.

The screenshot shows the 'Connect Account' page for ShinyApps.io. At the top, there's a header bar with the title 'Connect Account'. Below it, a section titled 'Connect Account' contains a blue cloud icon and the text 'ShinyApps.io' followed by a description: 'A cloud service run by RStudio. Publish Shiny applications and interactive documents to the Internet.' To the right of the text is a right-pointing arrow. Below this, another section titled 'Connect ShinyApps.io Account' has a back button and a 'Back' link. It includes instructions: 'Go to [your account on ShinyApps](#) and log in. Click your name, then choose **Tokens** from your account menu. Click **Show** on the token you want to use, then **Show Secret** and **Copy to Clipboard**. Paste the result here:' followed by a code block. The code block contains the following R code:

```
rsconnect::setAccountInfo(name='bioge',
token='46EBE9B38870523C47F96A9963ECF216',
secret='VfcFeIvText4xx54tbbxx1BstBkbioc6
RmfftAL1')
```

At the bottom of this section is a link 'Need a ShinyApps.io account? [Get started here.](#)'

Activity 7. Interacting with ChatGPT through API

1. Start a new R script file. Save a ChatGPT.R. <https://github.com/irudnyts/openai>

```
install.packages("remotes")
remotes::install_github("irudnyts/openai", ref = "r6")

Sys.setenv(
  OPENAI_API_KEY = 'sk-proj-XEFD_g86'
)

library(openai)
client <- OpenAI()
completion <- client$chat$completions$create(
  model = "gpt-3.5-turbo",
  messages = list(list("role" = "user", "content" = "What's up?"))
)

completion$choices[[1]]$message$content
```