**Week 4:**

**The first version:**

R version 4.3.2 (2023-10-31) -- "Eye Holes"

Copyright (C) 2023 The R Foundation for Statistical Computing

Platform: x86\_64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

**> read.csv("C:/Users/lipeng/Downloads/clinical.cases\_selection.2023-12-17.csv")**

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'C:/Users/lipeng/Downloads/clinical.cases\_selection.2023-12-17.csv': No such file or directory

**> read.csv("/Users/lipeng/Downloads/clinical.cases\_selection.2023-12-17.csv")**

demographic.cause\_of\_death demographic.race demographic.gender

1 Cancer Related white male

2 Cancer Related white male

3 white female

4 black or african american male

5 white male

6 white male

7 not reported male

8 white female

9 white male

10 white male

11 white male

12 white male

13 white male

14 white male

15 white male

16 Unknown white female

17 white male

18 asian male

19 asian male

20 white female

demographic.ethnicity demographic.vital\_status demographic.age\_at\_index

1 not hispanic or latino Dead 74

2 not hispanic or latino Dead 65

3 not hispanic or latino Alive 62

4 not hispanic or latino Alive 69

5 not hispanic or latino Alive 81

6 not reported Dead 68

7 not reported Alive 68

8 not hispanic or latino Alive 71

9 not hispanic or latino Alive 58

10 not hispanic or latino Alive 57

11 not hispanic or latino Alive 74

12 not hispanic or latino Dead 71

13 not hispanic or latino Alive 69

14 not hispanic or latino Alive NA

15 not reported Alive NA

16 not reported Dead NA

17 not reported Alive NA

18 not reported Not Reported NA

19 not reported Alive NA

20 not hispanic or latino Alive NA

demographic.submitter\_id demographic.days\_to\_birth

1 CDDP-A8B6\_demographic -27034

2 CDDP-ACO6\_demographic -24059

3 TCGA-55-8506\_demographic -22850

4 TCGA-L9-A7SV\_demographic -25298

5 TCGA-55-7994\_demographic -29858

6 TCGA-78-7155\_demographic -24863

7 TCGA-05-4382\_demographic -24868

8 TCGA-69-7979\_demographic -26180

9 TCGA-86-8073\_demographic -21214

10 TCGA-MN-A4N4\_demographic -20888

11 TCGA-18-3409\_demographic -27154

12 TCGA-21-1079\_demographic -26176

13 TCGA-90-A4ED\_demographic -25453

14 C3L-00144-DEMO -21487

15 C3N-00294-DEMO -21155

16 C3N-04176-DEMO -23173

17 C3L-03985-DEMO -24731

18 C3N-02152-DEMO -22161

19 C3N-00560-DEMO -20372

20 HCM-BROD-0027-C34\_demographic -23897

demographic.created\_datetime demographic.year\_of\_birth

1 2017-07-20T17:56:46.424041-05:00 1929

2 2017-07-20T18:00:31.412505-05:00 1938

3 1950

4 1944

5 1930

6 1937

7 1941

8 1940

9 1953

10 1953

11 1930

12 1924

13 1943

14 2017-10-30T14:12:23.237346-05:00 1957

15 2017-11-01T09:34:04.825788-05:00 1958

16 2019-06-24T07:45:49.778902-05:00 1955

17 2019-06-24T07:45:49.778902-05:00 1950

18 2019-06-24T07:45:49.778902-05:00 1957

19 2017-11-01T09:34:04.825788-05:00 1961

20 2019-04-04T15:11:28.695214-05:00 1950

demographic.cause\_of\_death\_source demographic.premature\_at\_birth

1 NA NA

2 NA NA

3 NA NA

4 NA NA

5 NA NA

6 NA NA

7 NA NA

8 NA NA

9 NA NA

10 NA NA

11 NA NA

12 NA NA

13 NA NA

14 NA NA

15 NA NA

16 NA NA

17 NA NA

18 NA NA

19 NA NA

20 NA NA

demographic.weeks\_gestation\_at\_birth demographic.demographic\_id

1 NA 7cea583d-41bc-561b-8536-2a2a373d4225

2 NA 2ce7d14e-0e86-5fae-a9b3-8fc2dc5307fd

3 NA 9a307bd1-a3de-570d-8093-a744620305e0

4 NA baf8d896-ff23-5478-b919-63bf1c0703db

5 NA fde1ca64-8a6a-5b99-b8bf-43dd3a4a0302

6 NA 5e9f31d2-5857-544b-8111-f754d64d1eb0

7 NA c2f0c511-8275-502d-b819-2405c10d0fbd

8 NA 708eacd3-4558-55a9-8fcd-836b599e9d63

9 NA a7f73687-5e40-53d9-a2d9-6512eb86f33d

10 NA 17131553-dff0-5b14-b75a-7a68fdbf575b

11 NA 276fefa4-778e-52fc-b57c-757da0761d85

12 NA 658c1601-2e5b-5d74-8474-a576e101d019

13 NA 8a6ffd75-9ace-51a1-bddc-9a93cfc38d06

14 NA 78282c0f-3581-4b2e-b3dd-1fef880f8541

15 NA afab402f-ff34-4120-a45c-a06c3e43f707

16 NA fd3dea48-0e0d-4cd2-b916-82225d464d8c

17 NA e81431fb-1813-4c7a-8026-6c0006e4d475

18 NA e3ec6c8f-d4df-4e97-9e86-bbe2d25f9b1e

19 NA abb77a27-e364-4242-a0ba-0e9525a908bb

20 NA 7dc95354-684a-4dd2-9ea8-f32985ba4049

demographic.updated\_datetime demographic.age\_is\_obfuscated

1 2022-11-04T14:22:25.013629-05:00

2 2022-11-04T14:22:25.013629-05:00

3 2019-07-31T20:25:06.838855-05:00

4 2019-07-31T20:27:16.945354-05:00

5 2019-07-31T20:28:39.818993-05:00

6 2019-07-31T20:43:55.452352-05:00

7 2019-07-31T20:10:29.538913-05:00

8 2019-07-31T20:36:03.858140-05:00

9 2019-07-31T20:39:30.743395-05:00

10 2019-07-31T20:39:40.508097-05:00

11 2019-07-31T19:23:45.322291-05:00

12 2019-07-31T19:22:32.552226-05:00

13 2019-07-31T19:19:59.261100-05:00

14 2019-06-24T07:45:49.778902-05:00

15 2023-02-03T17:42:19.917554-06:00

16 2023-02-03T17:42:19.917554-06:00

17 2023-02-03T17:42:19.917554-06:00

18 2023-02-03T17:42:19.917554-06:00

19 2023-02-03T17:42:19.917554-06:00

20 2020-10-08T15:36:07.819253-05:00 False

demographic.occupation\_duration\_years demographic.days\_to\_death

1 NA 857

2 NA 1647

3 NA NA

4 NA NA

5 NA NA

6 NA 1171

7 NA NA

8 NA NA

9 NA NA

10 NA NA

11 NA NA

12 NA 965

13 NA NA

14 NA NA

15 NA NA

16 NA 289

17 NA NA

18 NA NA

19 NA NA

20 NA NA

demographic.state demographic.year\_of\_death

1 released 2005

2 released 2007

3 released NA

4 released NA

5 released NA

6 released 2008

7 released NA

8 released NA

9 released NA

10 released NA

11 released NA

12 released 1997

13 released NA

14 released NA

15 released NA

16 released 2019

17 released NA

18 released NA

19 released NA

20 released NA

**> library(shiny)**

**>**

**> # Define UI**

**> ui <- fluidPage(**

**+ titlePanel("Clinical Cases Data Viewer"),**

**+**

**+ sidebarLayout(**

**+ sidebarPanel(**

**+ selectInput("selectedRace", "Select Race:", choices = NULL),**

**+ selectInput("selectedGender", "Select Gender:", choices = NULL),**

**+ selectInput("selectedStatus", "Select Vital Status:", choices = NULL)**

**+ ),**

**+**

**+ mainPanel(**

**+ dataTableOutput("filteredData")**

**+ )**

**+ )**

**+ )**

**> # Define server logic**

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

**+ # Load the dataset (make sure to adjust the file path)**

**+ data <- read.csv("/Users/lipeng/Downloads/clinical.cases\_selection.2023-12-17.csv")**

**+**

**+ # Update the choices for the select inputs**

**+ observe({**

**+ updateSelectInput(session, "selectedRace", choices = unique(data$demographic.race))**

**+ updateSelectInput(session, "selectedGender", choices = unique(data$demographic.gender))**

**+ updateSelectInput(session, "selectedStatus", choices = unique(data$demographic.vital\_status))**

**+ })**

**+**

**+ # Filter and display data based on selections**

**+ output$filteredData <- renderDataTable({**

**+ filteredData <- data[data$demographic.race == input$selectedRace &**

**+ data$demographic.gender == input$selectedGender &**

**+ data$demographic.vital\_status == input$selectedStatus, ]**

**+ if (nrow(filteredData) > 0) {**

**+ filteredData**

**+ } else {**

**+ data.frame()**

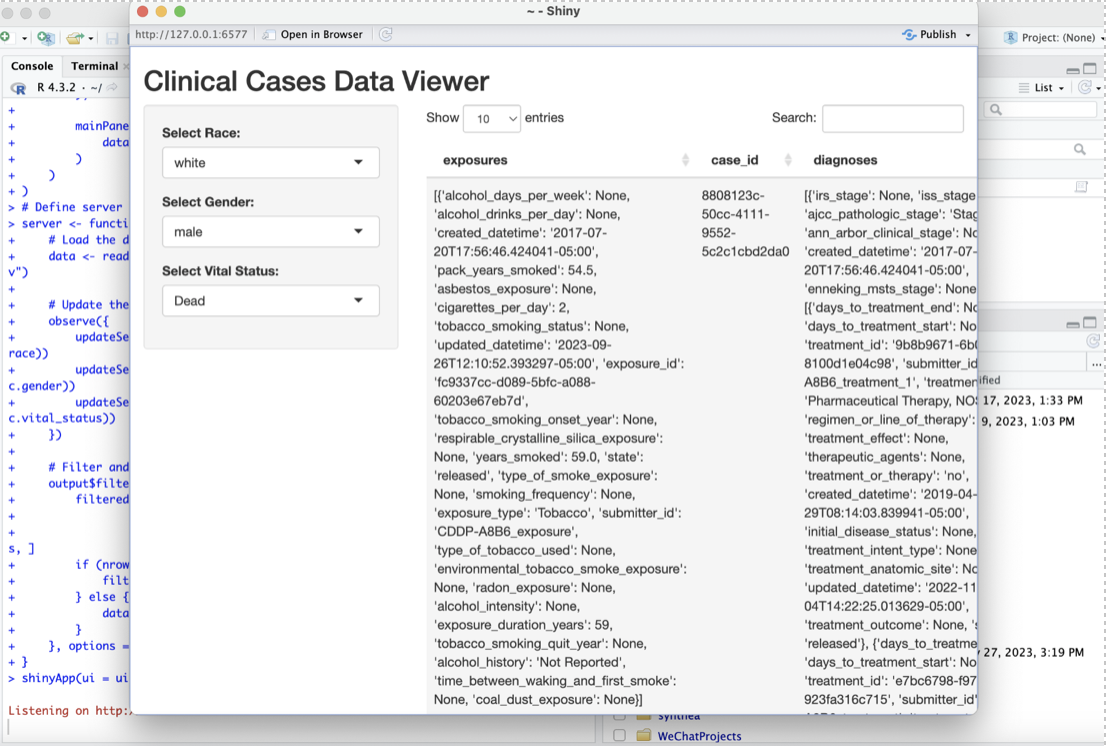
**+ }**

**+ }, options = list(pageLength = 10))**

**+ }**

**> shinyApp(ui = ui, server = server)**

**The result:**



Week 6:

Siavash provides a data portal about cancer (<https://portal.gdc.cancer.gov/>). I used R shiny to visualize the related skin cancer json files.

R version 4.3.2 (2023-10-31) -- "Eye Holes"

Copyright (C) 2023 The R Foundation for Statistical Computing

Platform: x86\_64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

> library(jsonlite)

> file\_path <- "/Users/lipeng/Downloads/cases.2023-01-11.json"

> json\_data <- fromJSON(file\_path)

Error: lexical error: invalid char in json text.

/Users/lipeng/Downloads/cases.20

(right here) ------^

> install.packages("jsonlite")

Error in install.packages : Updating loaded packages

Restarting R session...

> install.packages("jsonlite")

Error in install.packages : Updating loaded packages

Restarting R session...

> install.packages("jsonlite")

Error in install.packages : Updating loaded packages

> install.packages("jsonlite")

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-x86\_64/contrib/4.3/jsonlite\_1.8.8.tgz'

Content type 'application/x-gzip' length 1134427 bytes (1.1 MB)

==================================================

downloaded 1.1 MB

The downloaded binary packages are in

/var/folders/8c/ktg1k2g91w56srcc\_s6g86040000gn/T//RtmpknhN4U/downloaded\_packages

> print(json\_data)

Error: object 'json\_data' not found

> json\_data <- fromJSON(file\_path)

Error in fromJSON(file\_path) : could not find function "fromJSON"

> file\_path <- "/Users/lipeng/Downloads/cases.2023-12-14.json"

> json\_data <- fromJSON(file\_path)

Error in fromJSON(file\_path) : could not find function "fromJSON"

> library(jsonlite)

> file\_path <- "/Users/lipeng/Downloads/cases.2023-12-14.json"

> json\_data <- fromJSON(file\_path)

> print(json\_data)

**ui <- fluidPage(**

**titlePanel("Case Data Visualization"),**

**sidebarLayout(**

**sidebarPanel(**

**# Add input elements like selectInput or sliderInput here**

**),**

**mainPanel(**

**# Output: Plot or Table**

**plotOutput("plot") # or tableOutput("table")**

**)**

**)**

**)**

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

**# Reactive expressions to filter data based on input**

**# Render plot or table based on filtered data**

**output$plot <- renderPlot({**

**# Plotting code goes here**

**})**

**}**

**ShinyApp（UI = UI，Server = Server）**

**Week 8:**

*# Lists for different attributes*  
genders **=** ['Male', 'Female', 'Other', 'Prefer Not to Say']  
occupations **=** ['Engineer', 'Teacher', 'Doctor', 'Clerk', 'Chef', 'Unemployed']  
marital\_statuses **=** ['Single', 'Married', 'Divorced', 'Widowed']  
states **=** ['Victoria']  
postal\_codes **=** ['3000', '3004', '3008', '3010', '3015', '3025']  
country **=** "Australia"  
income\_ranges **=** ['<30,000', '30,000-60,000', '60,000-90,000', '90,000-120,000', '>120,000']

R version 4.3.2 (2023-10-31) -- "Eye Holes"

Copyright (C) 2023 The R Foundation for Statistical Computing

Platform: x86\_64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

*# Load libraries*

**> library(shiny)**

**>**

**> library(ggplot2)**

Error in library(ggplot2) : there is no package called ‘dplyr’

**> install.packages("ggplot2")**

**> install.packages("DT")**

**> install.packages("leaflet")**

**> library(DT)**

**Attaching package: ‘DT’**

The following objects are masked from ‘package:shiny’:

dataTableOutput, renderDataTable

**> library(ggplot2)**

**> library(leaflet)**

**#read csv**

**> read.csv("/Users/lipeng/Downloads/csv/patients.csv")**

**> # Define UI**

> ui <- fluidPage(

+ titlePanel("Patient Data Visualization"),

+ sidebarLayout(

+ sidebarPanel(

+ h3("Visualization Filters"),

+ selectInput("variable", "Select a variable to display:",

+ choices = c("Healthcare Expenses" = "HEALTHCARE\_EXPENSES",

+ "Healthcare Coverage" = "HEALTHCARE\_COVERAGE")),

+ sliderInput("yearRange", "Select Year of Birth Range:",

+ min = 1900, max = 2020, value = c(1980, 2000))

+ ),

+ mainPanel(

+ tabsetPanel(

+ tabPanel("Table", DTOutput("table")),

+ tabPanel("Map", leafletOutput("map")),

+ tabPanel("Histogram", plotOutput("histogram")),

+ tabPanel("Birth Year Distribution", plotOutput("birthYearPlot"))

+ )

+ )

+ )

+ )

# Define server logic

> server <- function(input, output) {

+ # Load data

+ data <- read.csv("/Users/lipeng/Downloads/csv/patients.csv")

+

+ # Render Table

+ output$table <- renderDT({

+ datatable(data)

+ })

# Render Map

+ output$map <- renderLeaflet({

+ leaflet(data) %>%

+ addTiles() %>%

+ addCircleMarkers(lng = ~LON, lat = ~LAT, popup = ~as.character(FIRST))

+ })

+

+ # Render Histogram

+ output$histogram <- renderPlot({

+ ggplot(data, aes\_string(x = input$variable)) +

+ geom\_histogram(binwidth = 1000, fill = "blue", color = "black") +

+ theme\_minimal() +

+ xlab(input$variable) +

+ ylab("Count")

+ })

# Render Birth Year Distribution

+ output$birthYearPlot <- renderPlot({

+ birthYear <- as.numeric(format(as.Date(data$BIRTHDATE), "%Y"))

+ filteredData <- data[birthYear >= input$yearRange[1] & birthYear <= input$yearRange[2],]

+ ggplot(filteredData, aes(x = as.factor(as.numeric(format(as.Date(BIRTHDATE), "%Y"))))) +

+ geom\_bar() +

+ theme\_minimal() +

+ xlab("Year of Birth") +

+ ylab("Number of Patients")

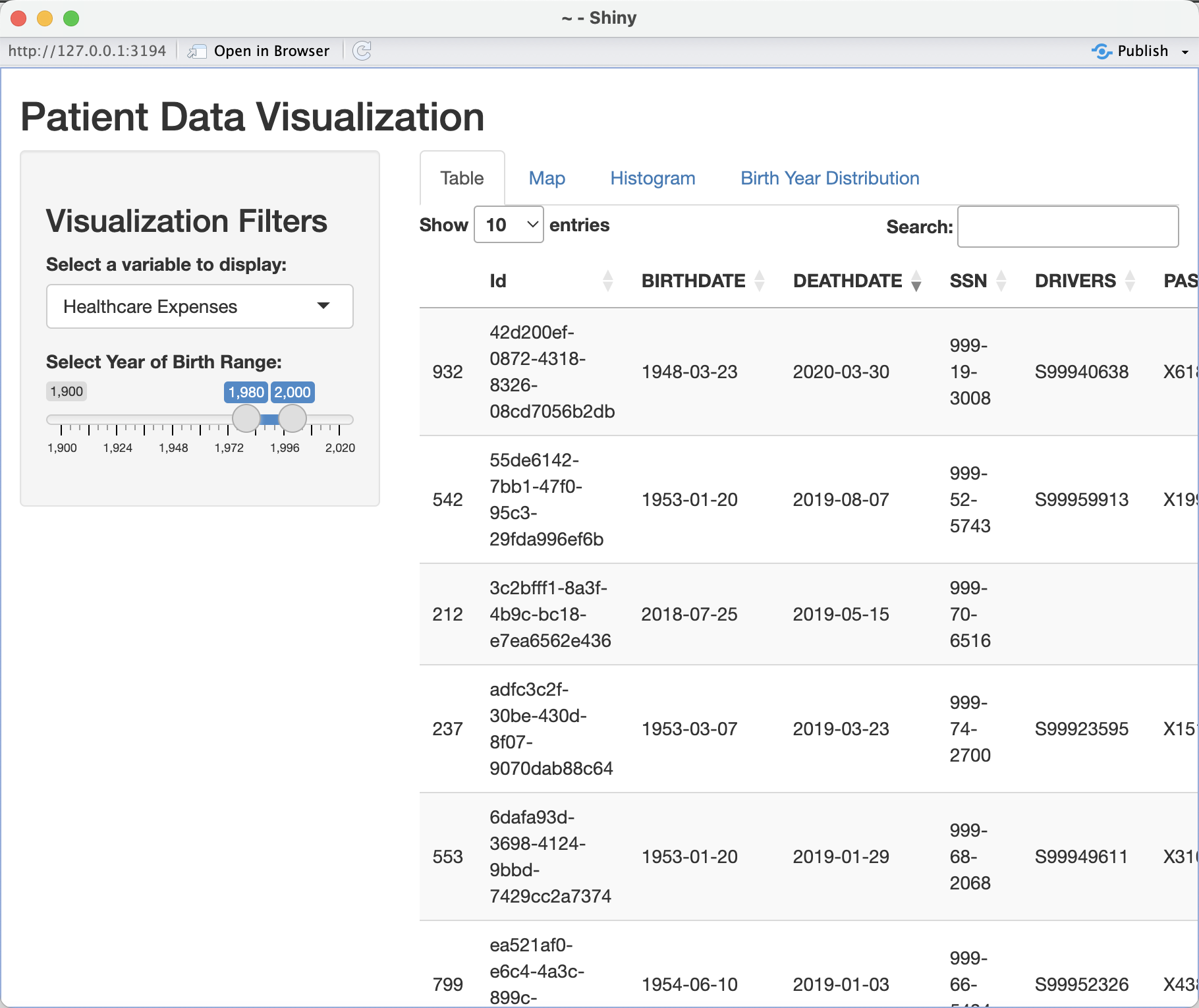
+ })

+ }

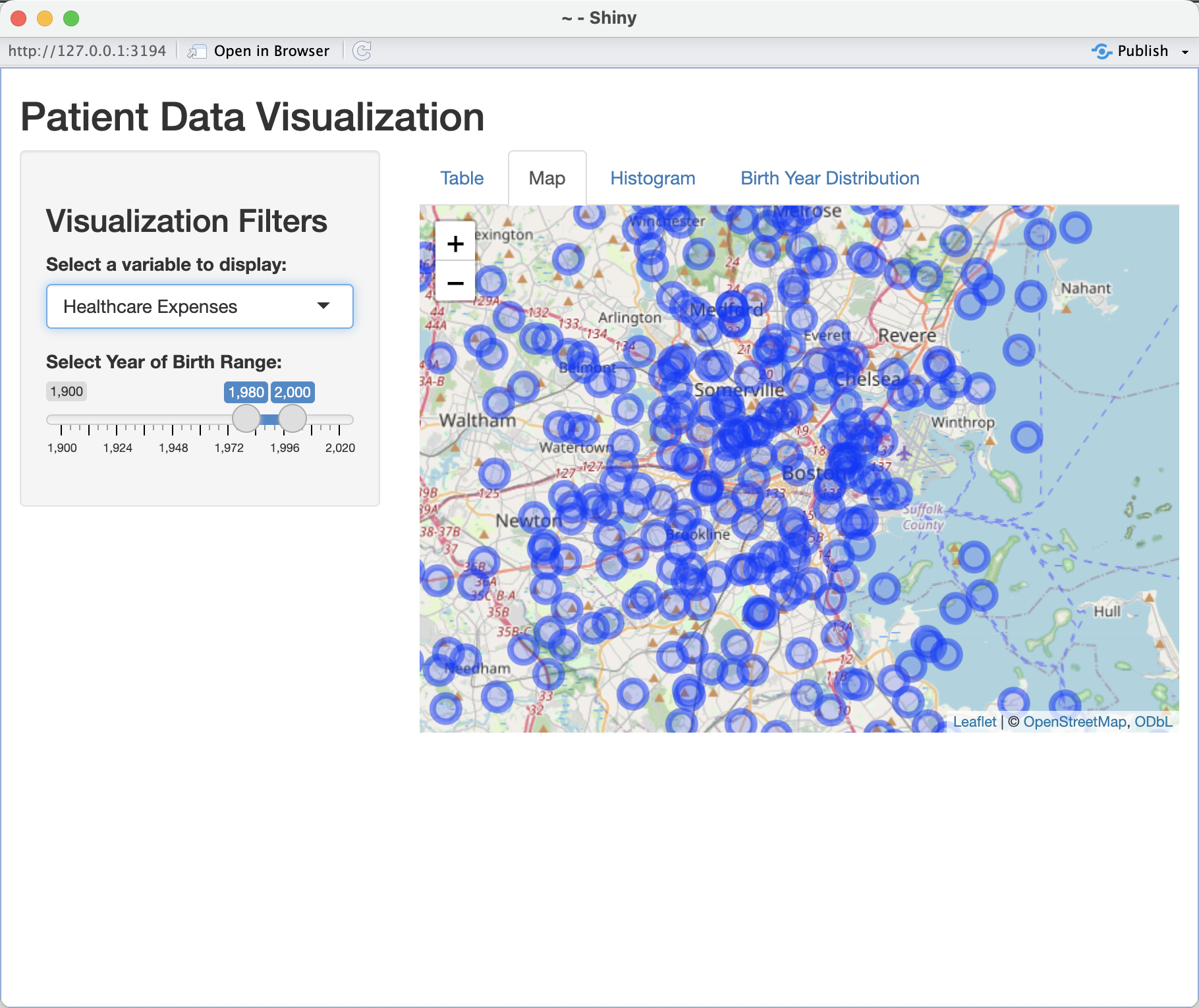
# Run the application

> shinyApp(ui = ui, server = server)

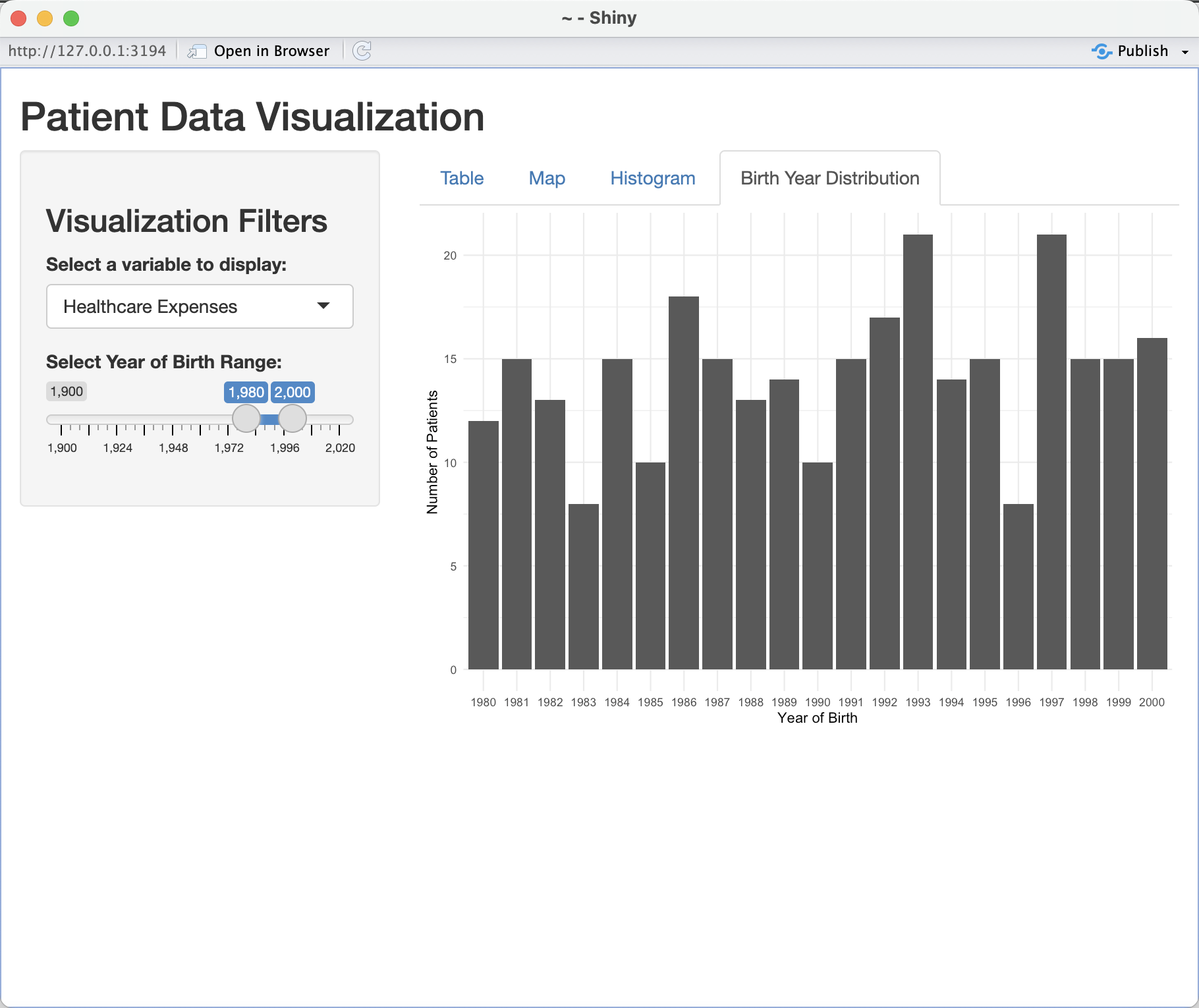
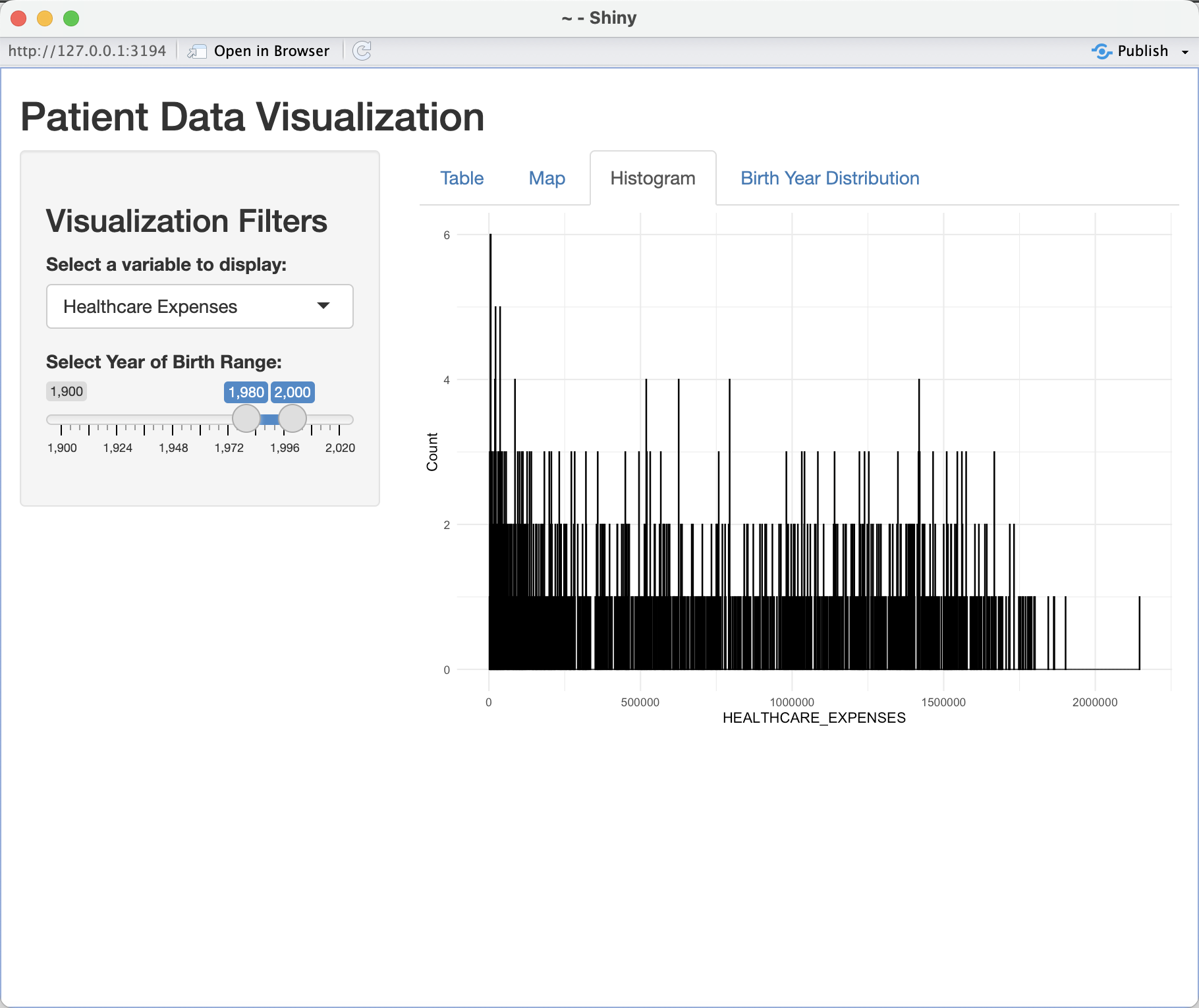
**The result:**



There is also specific case information at the back, but I can’t capture it all in the screenshot.



There are too many samples, so it will be messy when zoomed out.



Week 11:

The dataset I used was obtained from GDC center's [Lung cancer.](https://portal.gdc.cancer.gov/analysis_page?app=)

**It is the code for Clinical.tsv:**

library(shiny)

library(ggplot2)

# Define UI

ui <- fluidPage(

titlePanel("Clinical Data Visualization"),

sidebarLayout(

sidebarPanel(

# Input: Selector for different visualizations

selectInput("visualization", "Choose a visualization:",

choices = c("Age Distribution", "Project ID Distribution")),

# Additional options can be added here

),

mainPanel(

plotOutput("plot"), # Output: Plot

tableOutput("table") # Output: Data table

)

)

)

# Define server logic

server <- function(input, output) {

# Reactive expression to read and process the data

processedData <- reactive({

data <- read.csv("path/to/your/clinical.tsv", sep = "\t")

# Data pre-processing steps go here

data

})

# Generate plot based on input selection

output$plot <- renderPlot({

data <- processedData()

if (input$visualization == "Age Distribution") {

# Assuming 'age\_at\_index' is the column for age

ggplot(data, aes(x = as.numeric(age\_at\_index))) +

geom\_histogram(binwidth = 5, fill = "blue", color = "black") +

theme\_minimal() +

xlab("Age") + ylab("Number of Cases") +

ggtitle("Distribution of Cases by Age")

} else if (input$visualization == "Project ID Distribution") {

ggplot(data, aes(x = factor(project\_id))) +

geom\_bar(fill = "green", color = "black") +

theme\_minimal() +

xlab("Project ID") + ylab("Number of Cases") +

ggtitle("Distribution of Cases by Project ID") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

}

})

# Display a subset of the data as a table

output$table <- renderTable({

data <- processedData()

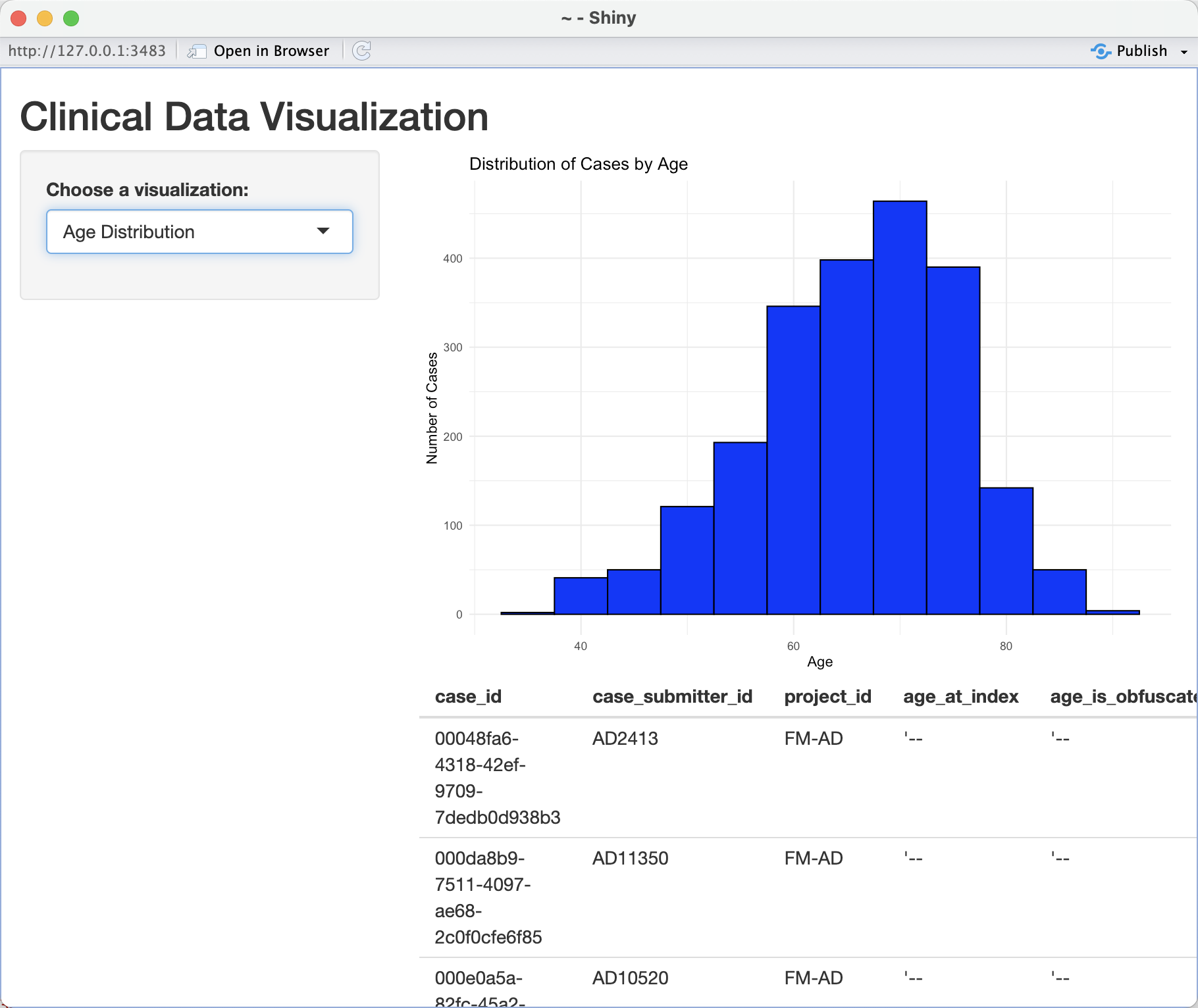
head(data) # Show only the first few rows

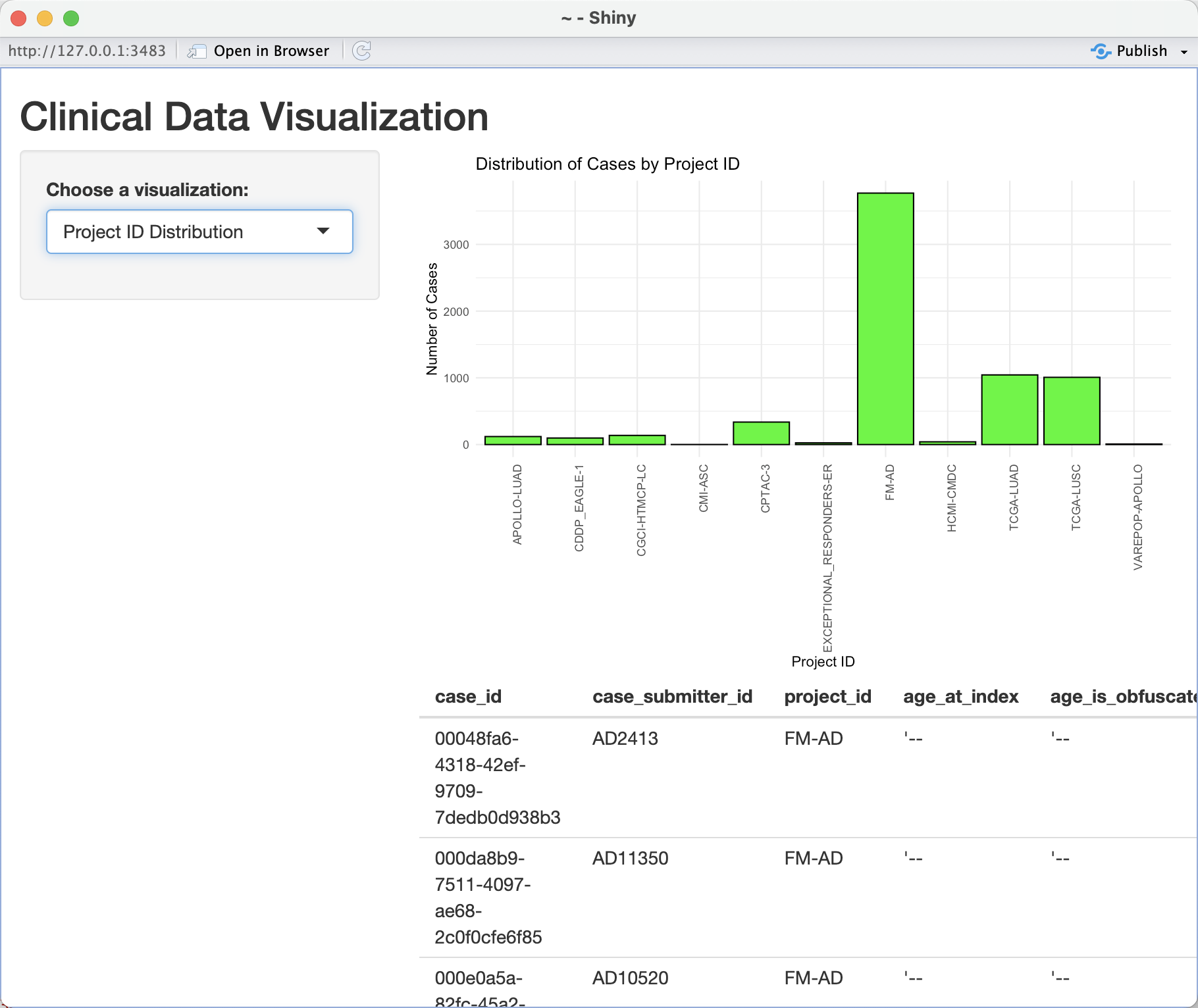
})

}

# Run the application

shinyApp(ui = ui, server = server)

The Result:



**Here is the code for exposure.tsv:**

library(shiny)

library(ggplot2)

# Define UI

ui <- fluidPage(

titlePanel("Exposure Data Visualization"),

sidebarLayout(

sidebarPanel(

# Input: Selector for different visualizations

selectInput("visualization", "Choose a visualization:",

choices = c("Alcohol Intensity Distribution", "Tobacco Smoking Status Distribution", "Years Smoked Distribution")),

# Additional options can be added here

),

mainPanel(

plotOutput("plot"), # Output: Plot

tableOutput("table") # Output: Data table

)

)

)

# Define server logic

server <- function(input, output) {

# Reactive expression to read and process the data

processedData <- reactive({

data <- read.csv("path/to/your/exposure.tsv", sep = "\t")

# Data pre-processing steps go here

data

})

# Generate plot based on input selection

output$plot <- renderPlot({

data <- processedData()

if (input$visualization == "Alcohol Intensity Distribution") {

ggplot(data, aes(x = alcohol\_intensity)) +

geom\_bar(fill = "blue", color = "black") +

theme\_minimal() +

xlab("Alcohol Intensity") + ylab("Number of Cases") +

ggtitle("Distribution of Alcohol Intensity") +

theme(axis.text.x = element\_text(angle = 45, hjust = 1))

} else if (input$visualization == "Tobacco Smoking Status Distribution") {

ggplot(data, aes(x = tobacco\_smoking\_status)) +

geom\_bar(fill = "green", color = "black") +

theme\_minimal() +

xlab("Tobacco Smoking Status") + ylab("Number of Cases") +

ggtitle("Distribution of Tobacco Smoking Status") +

theme(axis.text.x = element\_text(angle = 45, hjust = 1))

} else if (input$visualization == "Years Smoked Distribution") {

# Assuming 'years\_smoked' can be treated as numeric

ggplot(data, aes(x = as.numeric(years\_smoked))) +

geom\_histogram(binwidth = 5, fill = "red", color = "black") +

theme\_minimal() +

xlab("Years Smoked") + ylab("Number of Cases") +

ggtitle("Distribution of Years Smoked")

}

})

# Display a subset of the data as a table

output$table <- renderTable({

data <- processedData()

head(data) # Show only the first few rows

})

}

# Run the application

shinyApp(ui = ui, server = server)

The Result: 