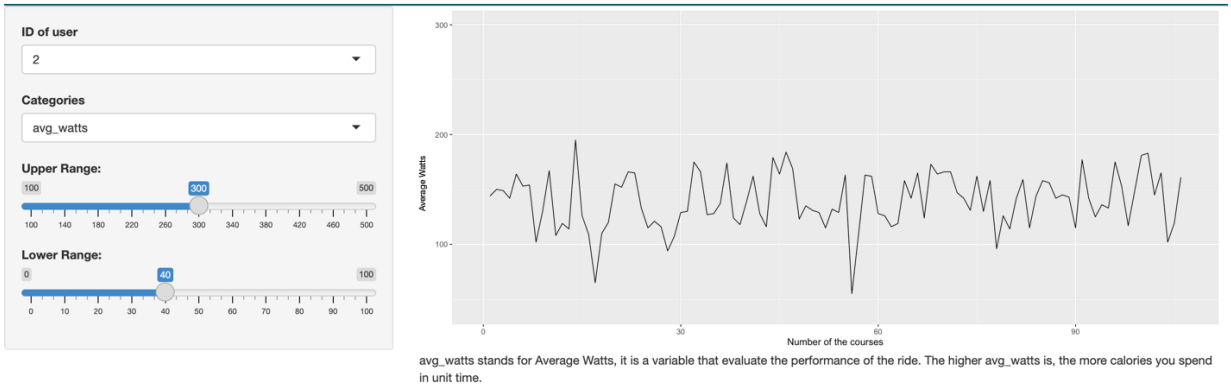
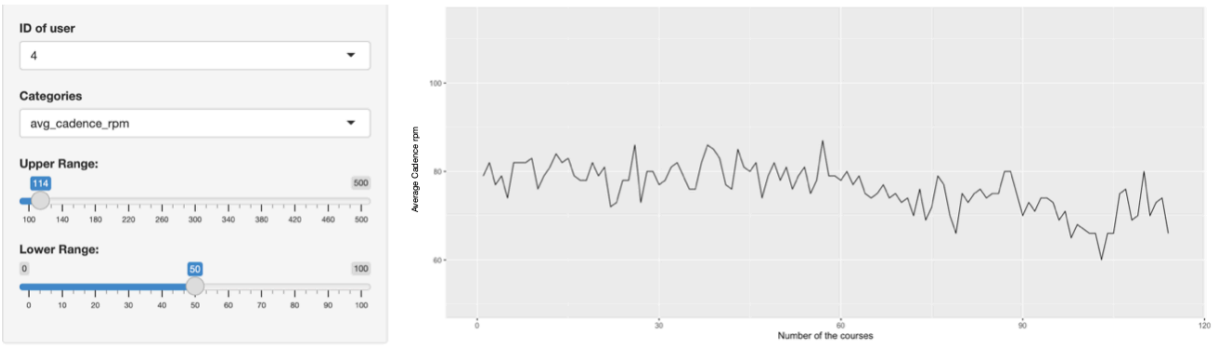
Perspective Of Data Science HW4 Yucheng Wang
Question 1,



avg_watts stands for Average Watts, it is a variable that evaluate the performance of the ride. The higher avg_watts is, the more calories you spend in unit time.



avg_cadence_rpm stands for the Average cadence rotation per minute, it shows the average speed in the ride.

Here is an interactive shiny app of the cycling records, by choosing the User id, you can choose the records of different users, and by choosing different categories, you can see average watts, average cadence rpm and calories burned for each person.

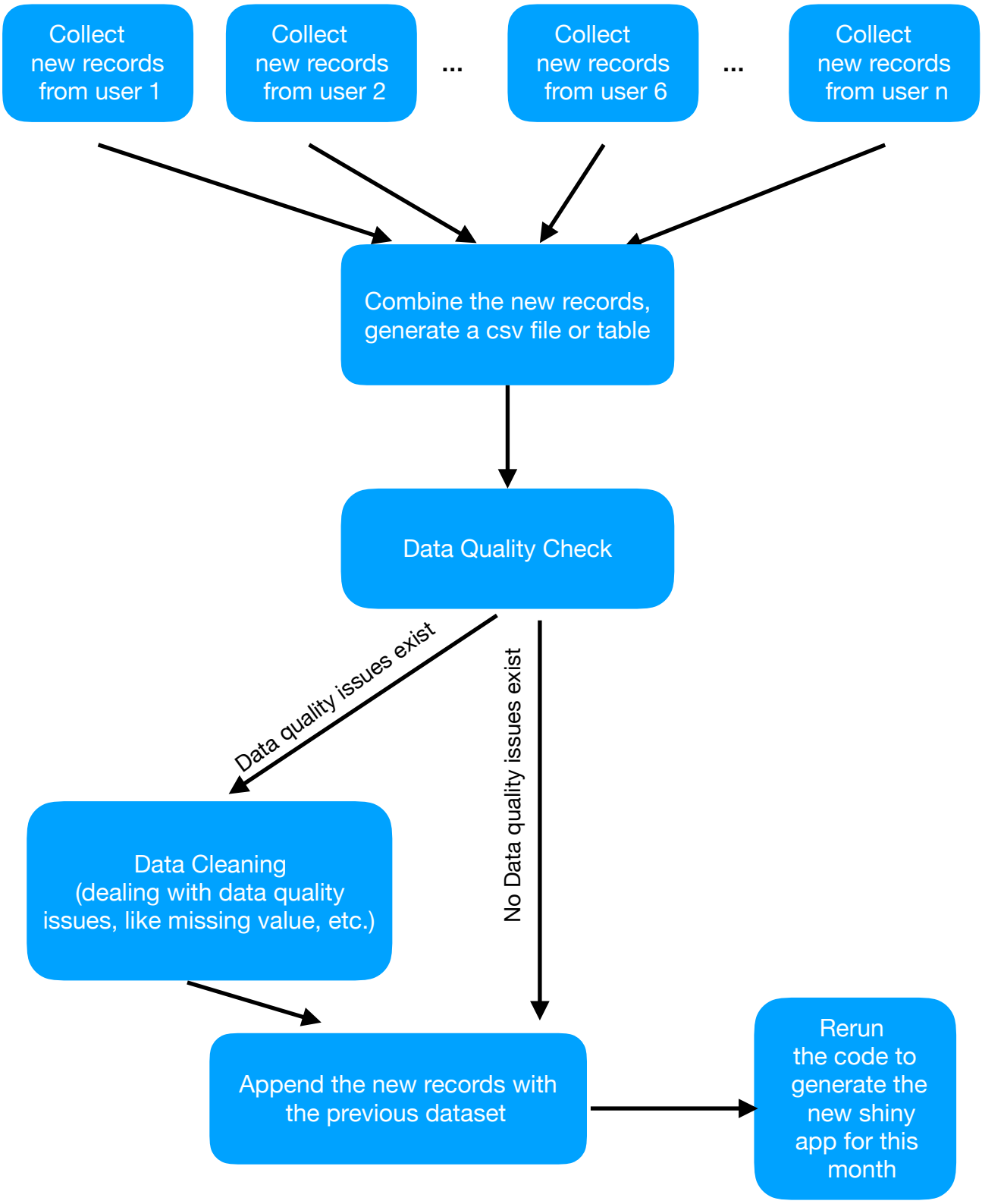
Code:

```
library(shiny)
library(gapminder)
#help(gapminder)
library(tidyverse)
all_workouts <- read_csv("/Users/wyc/all_workouts.csv")
all_workouts<-as_tibble(all_workouts)
#help(textInput)
ui <- fluidPage(
  sidebarLayout(
    sidebarPanel(
      selectInput(inputId = "person_id", label = "ID of user",
                  choices = c("1", "2", "3","4", "5", "6"),
                  selected = "1"),
      selectInput(inputId = "charts", label = "Categories",
                    choices = c("avg_watts", "calories_burned", "avg_cadence_rpm"),
```

```
                selected = "avg_watts"),
    sliderInput("Range1", "Upper Range:",
                min = 100, max = 500,
                value = 300),
    sliderInput("Range2", "Lower Range:",
                min = 0, max = 100,
                value = 40)
  ),
  mainPanel(
    plotOutput(outputId = "tsplot"),
    textOutput(outputId = "selected_var")
  )
 )
)


server <- function(input, output) {
 output$tsplot <- renderPlot({

  datafiltered <- all_workouts %>%
    filter(person_id == input$person_id)
  ggplot(datafiltered, aes_string("class_counter", input$charts))+
    geom_line(size = 0.4) +
    xlab("Number of the courses")+
    geom_col() +
    scale_y_continuous(limits = c(input$Range2,input$Range1)) +
    scale_fill_discrete()

 })
 output$selected_var = renderText(switch(input$charts,avg_watts="avg_watts stands for
Average Watts, it is a variable that evaluate the performance of the ride. The higher avg_watts
is, the more calories you spend in unit time.",calories_burned = "calories_burned stands for the
total calories you burned in the ride.",avg_cadence_rpm="avg_cadence_rpm stands for the
Average cadence rotation per minute, it shows the average speed in the ride."))

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

Question 2,

Monthly workflow diagram for monthly process



Question 3,

The nodes in the diagram are the task we might need to implement each month to get our shiny apps with data updated. Generally, we do not need to modify our code each month, so the most important issue is to update data. Firstly, we need to collect the monthly new data from the users, and we should combine all those data together into a csv file, or we can also store them in a table in database. The raw data we obtained might not be usable and might have some data quality issues, for instance, the data format might not be correct and there might be a lot of missing values. So we need to proceed a data quality check to make sure the new data could meet our requirement. There could be two possible outcomes of the data quality check, if some quality issues are found, we need to do the data cleaning, and we might also need to reformat the new data, else, we might not need to clean the data. After that, we could combine the new data with the previous data and obtain the new data set. In the end, we could just rerun the code and our shiny app could be updated.

It is a good solution, most importantly, we do not need to change anything about our source code, the only thing we need to do is append the new data into our previous dataset and rerun the code, it is very convenient and efficient. Also, we will do thorough data quality check and data cleaning process each month to make sure the quality of the new data, which ensure the performance of our shiny app. Further, our new data would not influence the previous data, which means risk could be avoid that if any errors occur in our new data, our shiny app could still work functionally with the previous data. Thus, this diagram provides a convenient, efficient, high-performance and low-risk solution for the monthly update, which is the best.