```r
#Lines 5 through 20 are examples of various file types
#and the code to read and write them.
#Your tasks begin at line 22.

#Getting and saving your dataset is typically a two step process
#Read and write a delimited text file.
#datasetname <- read.table('file.txt')
#write.table(datasetname, 'file.txt')

#Read and write a comma separated value file. This is a special case
 of read.table/ write.table.
#datasetname <- read.csv('file.csv')
#write.csv(datasetname, 'file.csv')

#Read and write an R data file, a file type special for R.
#load('file.RData')
#save(datasetname, file = 'file.Rdata')

#Read and write an R data file from GitHub.
#You need to select 'raw data' on the GitHub page
#and then copy the URL and put in your code, as below

#TASK: run the code below to get and save the dataset
download.file(url = "https://projects.fivethirtyeight.com/soccer-
 api/international/2022/wc_matches.csv", destfile = "WorldCup.csv")
#Then you need to name your dataset. Run this:
WorldCup<- read.csv("WorldCup.csv")

#TASK: take a look at the World Cup data.
> WorldCup
or
> head(WorldCup)
or
> tail(WorldCup)

#TASK: Install and call the dplyr package.
> install.packages("dplyr")
> library("dplyr")

#Let's make a random sample of our data and save it
#Task: run the code below
mysample<-sample_n(WorldCup, size=15, replace = FALSE, weight =
 NULL, .env = NULL)

#TASK: Save the new sample as a csv file
```

```
> write.csv(mysample, file="mysample.csv", row.names=FALSE)

#Now let's have some fun with *piping*

#we will use our mysample dataset
#The pipe, %>%, comes from the magrittr package.
#Packages in the tidyverse (like dplyr) load %>% for you
 automatically,
#so you don't usually load magrittr explicitly.

#Example: Let's try some piping with our mysample data. Note how the
 dataset name is not repeated in each function
piping<-mysample %>%
  rename(SoccerPowerIndex = spi1) %>%
  subset(SoccerPowerIndex >60) %>%
  dim()%>%
  print()

#TASK: revise this code chunk using piping
mysample2<-mysample
arrange(mysample2, date)
mysample2<-filter(mysample2, spi1<80)
mysample2<-rename(mysample2, Index1 = spi1, Index2 = spi2)
mysample3<-select(mysample2, Index1, Index2, team1, team2 )
mysample4<-summary(mysample3)
print(mysample4)

piping:
mysample %>%
    arrange(date) %>%
    filter(spi1<80) %>%
    rename(Index1 = spi1, Index2 = spi2) %>%
    select(Index1, Index2, team1, team2) %>%
    summary() %>%
    print()
```