Homework 2 - R Intro

## R Intro

Fill out [this Quarto document](https://github.com/BIFX552-22/CourseInfo/hw/02Rintro.qmd) while completing the R Intro lesson.

### [Introduction to R and RStudio](https://swcarpentry.github.io/r-novice-gapminder/01-rstudio-intro/index.html)

|  |
| --- |
| Key Points |
| * Use RStudio to write and run R programs. * R has the usual arithmetic operators and mathematical functions. * Use <- to assign values to variables. * Use ls() to list the variables in a program. * Use rm() to delete objects in a script. * Use install.packages() to install packages and library() to load them. |

#### Challenge 1

Which of the following are valid R variable names?

min\_height  
max.height  
\_age  
.mass  
MaxLength  
min-length  
2widths  
celsius2kelvin

* Valid
  + min\_height
  + max.height
  + .mass
  + MaxLength
  + celsius2kelvin
* Invalid
  + \_age
  + min-length
  + 2widths

#### Challenge 2

What will be the value of each variable after each statement in the following program?

mass <- 47.5  
age <- 122  
mass <- mass \* 2.3  
age <- age - 20  
  
# print the results  
mass

[1] 109.25

age

[1] 102

#### Challenge 3

Run the code from the previous challenge, and write a command to compare mass to age. Is mass larger than age?

# comparison code goes here  
mass > age

[1] TRUE

#### Challenge 4

Clean up your working environment by deleting the mass and age variables.

# code goes here  
rm(mass, age)  
ls()

character(0)

#### Challenge 5

Install the following packages: ggplot2, dplyr, gapminder

install.packages(‘ggplot2’, ‘dplyr’, ‘gapminder’) #they have been installed already

# this will fail if the required packages are not installed  
library(ggplot2)  
library(dplyr)

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':  
  
 filter, lag

The following objects are masked from 'package:base':  
  
 intersect, setdiff, setequal, union

library(gapminder)

### [Project Intro](https://swcarpentry.github.io/r-novice-gapminder/02-project-intro/index.html)

|  |
| --- |
| Key Points |
| * Use RStudio to create and manage projects with consistent layout. * Treat raw data as read-only. * Treat generated output as disposable. * Separate function definition and application. |

#### Challenge 1

Create a self-contained project

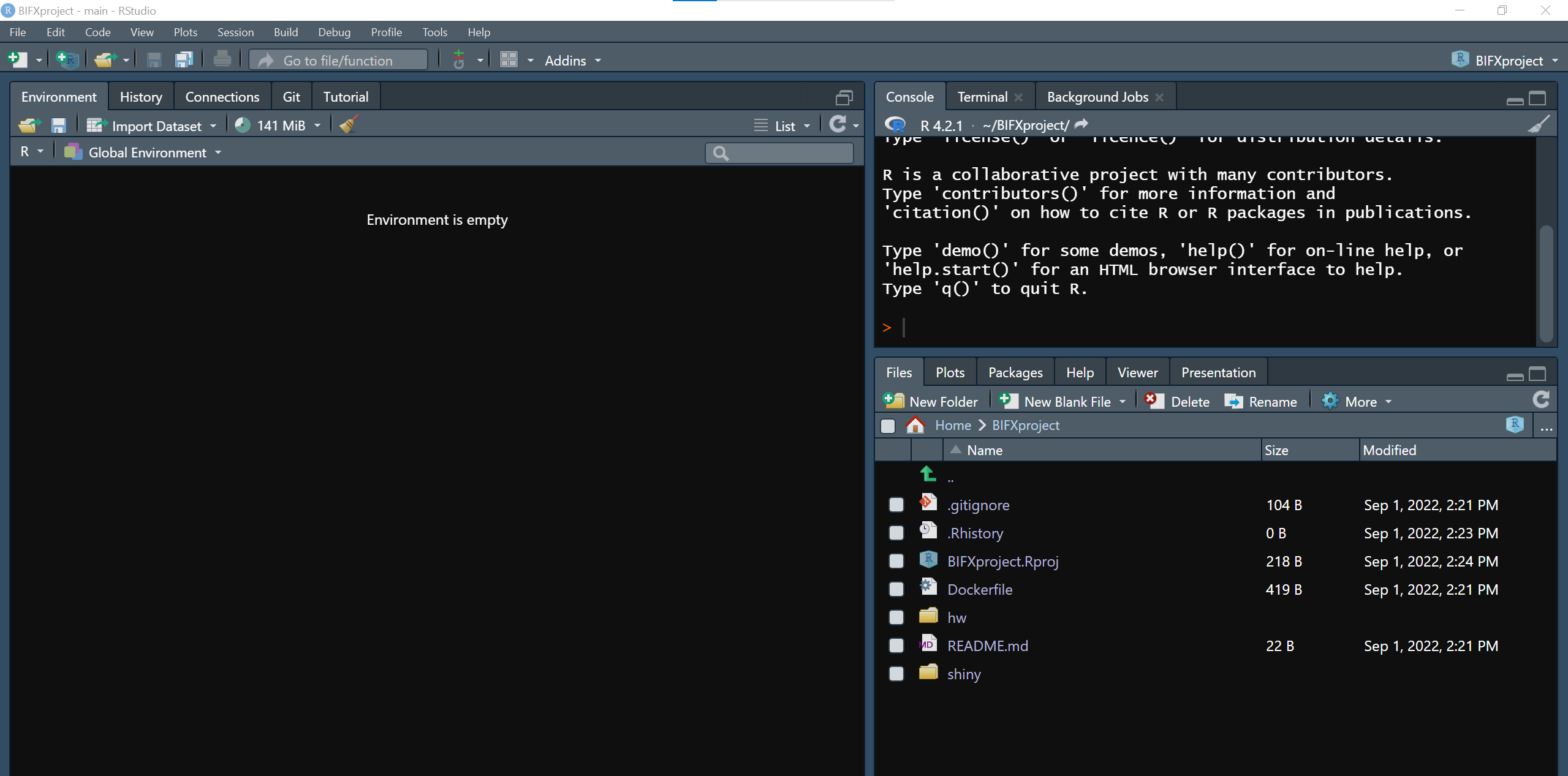
We’re going to turn our course project into a project in RStudio:

1. If you haven’t already, clone your course project onto your local machine.
2. In RStudio, click the “File” menu button, then “New Project”.
3. Click “Existing Directory” and locate the directory containing your course project.
4. Click the “Create Project” button.

#### Challenge 2

Open an RStudio project through the file system

1. Exit RStudio.
2. Navigate to your course project directory.
3. Create a new directory titled hw/ and save this file into that directory.
4. Double click the .Rproj file in the rood directory of your course project repository.
5. Open this file in RStudio.
6. Take a screenshot of your RStudio session and place it in the hw/img/ directory. It should have “BIFXproject - main - RStudio” in the header (or something similar).
7. Modify the line just below this to include your screenshot.



#### Challenge 3

You should have installed the gapminder package. If you are following along with the Software Carpentry notes, they ask us to download the gapminder data set at this point. Instead, we’ll load it directly from the gapminder package (i.e. you get this challenge for free).

library(gapminder)  
data(gapminder) # load the data directly from the 'gapminder' package  
  
# let's take a look at the data we'll be working with  
gapminder

# A tibble: 1,704 × 6  
 country continent year lifeExp pop gdpPercap  
 <fct> <fct> <int> <dbl> <int> <dbl>  
 1 Afghanistan Asia 1952 28.8 8425333 779.  
 2 Afghanistan Asia 1957 30.3 9240934 821.  
 3 Afghanistan Asia 1962 32.0 10267083 853.  
 4 Afghanistan Asia 1967 34.0 11537966 836.  
 5 Afghanistan Asia 1972 36.1 13079460 740.  
 6 Afghanistan Asia 1977 38.4 14880372 786.  
 7 Afghanistan Asia 1982 39.9 12881816 978.  
 8 Afghanistan Asia 1987 40.8 13867957 852.  
 9 Afghanistan Asia 1992 41.7 16317921 649.  
10 Afghanistan Asia 1997 41.8 22227415 635.  
# … with 1,694 more rows

### [Seeking Help](https://swcarpentry.github.io/r-novice-gapminder/03-seeking-help/index.html)

|  |
| --- |
| Key Points |
| Use help() or ? to get help in R. |

#### Challenge 1

Look at the help page for c. What kind of vector do you expect will be creeated if you evaluate the following?

c(1, 2, 3) # numeric vector

[1] 1 2 3

c('d', 'e', 'f') #character vector

[1] "d" "e" "f"

c(1, 2, 'f') #character vector

[1] "1" "2" "f"

#### Challenge 2

Look at the help for the paste function. You will need to use it later. What’s the difference between the sep and collapse arguments?

* ‘sep’ argument is a character string to separate the terms.
* ‘collapse’ argument is an optional character string to separate the results.

#### Challenge 3

Use help to find a function (and its associated parameters) that you could use to load data from a tabular file in which columns are delimited with “ (tab) and the decimal point is a”.” (period). This check for decimal separator is important, especially if you are working with international colleagues, because different countries have different conventions for the decimal point (i.e. comma vs period).

* read.delim(file, header = TRUE, sep = “, quote =”"“, dec =”.”, …)

|  |
| --- |
| Hint |
| Use ??"read table" to look up functions related to reading in tabular data. |