# Introduction To R

**Introduction lab worksheet**

We invite people to contribute to a website (https://whyiuser.wordpress.com/) where they explain how they use R, how they got started with R, and what the like/dislike about it. Check out the website and read at least two entries.

1. What are two things that you would like to be able to do with R that will be useful for your school/career plans?

2. What are your concerns about learning R?

**RStudio**

First, you should get up and running with RStudio. You can do this on classroom computers by opening the application. If you have a personal computer, download and install RStudio (free!).

- RStudio installation tutorial for Windows: https://www.youtube.com/watch?t=2&v=5ZbjUEg4a1g
- RStudio installation tutorial for Mac: https://www.youtube.com/watch?t=9&v=buCEFFuLpYo
  Watch these Udacity mini videos to learn about
- RStudio basics: https://www.youtube.com/watch?&v=FDSmIlBy7ko
- adjusting RStudio settings: https://www.youtube.com/watch?&v=vLIj5nNj8x4
- getting help with R: https://www.youtube.com/watch?v=ABVX527RODE
  A couple additional time-saving tips:
- when you're in the console, you can use the up arrow to go through your command history
- to execute a line of code from your script in the console, put the cursor anywhere on that line and type command enter

3. Complete the exercise showing which panel can take care of each function at the end of the RStudio basics video (this may be easier to come back to after you've used RStudio a bit)

a. Save an R script

b. Review the log of commands entered

c. Read help documentation

d. Clear the workspace

e. Run a bunch of commands from a file

f. Look at a plot

g. See a list of objects in memory

h. Read the results from functions or calculations

**Learning R**

There are tons of great resources on line to help you learn the basics of R. Here's a few that might be useful

- CodeSchool (extensive interactive intro to R): http://tryr.codeschool.com/levels/1/challenges/1
- DataCamp (another extensive interactive intro to R): https://www.datacamp.com/getting-started?step=2&track=r
- R tutorial (more of a textbook layout): http://www.r-tutor.com/r-introduction
- R for cats (campy. Particularly intro is useful): http://rforcats.net/

## Expressions

While R is great to get complicated statistical analyses done easily, it can also do more simple math. Try entering simple math commands in the console (like you would with a graphing calculator).

4. What expression do you enter into the console to get
   a. the sum of 4 and 11?
   b. the product of 9 and 7?
   c. 0.83 to the 10th?
   d. the decimal form of the ratio 5 to 7?
5. Think of a simple real-world math question (calculating tip, splitting a bill, etc).
   a. Write down your question in one sentence.
   b. Solve it yourself using R.
   c. Trade questions with a partner and solve each other's.
   d. Check to see if your answers are the same, if not, discuss.

## Variables

Variables are used to store values. There are two ways to assign variables:

```
> x = 3
```

or

```
> x <- 3
```

6. Practice making variables as
   a. save the value 3.14159 to a variable named `pi`
   b. compute the sum of 1, 2, 3, and 4, saving the value to the variable `sum`
   c. save the value 7 to a variable named `r`
   d. compute twice the product of `pi` and `r`, saving the value to the variable `circumference`

Note how the variables you create and their values appear in the environment window of RStudio. You can also see the value of a variable in the console by typing its name.

7. Use the console to check the values of the variables you created
   a. `pi`
   b. `sum`
   c. `r`
   d. `circumference`
8. In this class, we have article discussions lead by a randomly selected student. We'll walk through a calculation of the chance that you will be the student leader next week.
   a. Create a variable to store the number of students and set it to an appropriate value.
   b. Since you are one student, the chance that you will be the student leader is one divided by the total number of students. Using the variable you just created, compute that probability that you will be next week's leader.
   c. Think of a group of students that we counted (like students whose favorite color is purple). Create a variable representing the number of students in that group. Using the variable you created in (a) and this variable, compute the probability that a student in that group will be next week's leader.
   d. Compare your results with a neighbor.

**Scripts**

So far you've been using the console to interact with R. For projects, it's more practical to save R code in a script which you can modify and run many times. Create a new script in R (under the file menu -> new file -> R script, or icon in the top left corner).

Copy some of the commands you've already run into the script. Clear your environment of the variables you've made so far (broom icon at the top of the environment window). Experiment with running the whole thing (source icon in top of the script window, use source with echo to get the usual feedback in the console).

Clear your environment of variables again. Now experiment with running your script one line at a time (run icon at the top of the script window).

Tip: with your cursor on the line you'd like to run in the script window, you can simply type control-enter to run that line and advance the cursor to the next line.

**Comments**

Comments are useful for annotating your code so that you and other people can easily understand it. A comment is any text after a pound sign. That text is ignored by the R compiler. For example,

```
>x=3     # this line of code sets the value of x to 3
```

9. Use comments to annotate the variables as you created in question 5.

**Logical operations**

Logical operations are an automated way to check the value of a variable. From above, the variable x is set to the value 3, then we can use logical operators like this:

```
>x==3     # x equals 3?
[1] TRUE
> x < 3    # x is less than 3?
[1] FALSE
> x >= 3    # x is greater than or equal to 3?
[1] TRUE
```

10. Use logical operators to check if
    a. `pi` equals 3.14
    b. `pi` equals 3.14159
    c. `circumference` is greater than or equal to 40
    d. `circumference` is less than `r`

Logical variables can be combined with logic operators. For example, "A AND B" is true only when both A and B are true. In the console, this looks like

```
> (x == 3) && (pi > 3)
[1] TRUE
```

11. Use logical operators to check
    a. if the variables `x` and `pi` are both set to the value `3`
    b. if neither of the variables `x` and `pi` are set to the value `4`
    c. if the variable `x` is not `3` and `pi` is greater than `3`
    d. if the variable `x` is not `3` and `pi` is `3`

Another very useful logical operator is OR. "A OR B" is true when either A or B is true.

12. If we replace `&&` with `||` (the R symbol for OR) in the lines above, what do you expect the outcomes to be? After predicting the outcomes, use the console to confirm your predictions
    a. `> (x == 3) || (pi > 3)`
    b. `> (x == 3) || (pi == 3)`
    c. `> (x != 4) || (pi != 4)`
    d. `> (x != 3) || (pi > 3)`
    e. `> (x != 3) || (pi == 3)`