Use the notes provided to complete this assignment. You will need to learn this material to complete assignments in the future, so ask lots of questions / get them to help you BUT do not just ask for the answer).

You will turn in comments and the script created in this assignment on Blackboard. This assignment is graded complete/incomplete.

Questions:

Basic Math and Commands:

1. Try adding more than two numbers.
2. Try playing around with the number of spaces between numbers and the plus sign. Do the number of spaces matter?
3. Try hitting enter when you are half way through the command (e.g., 10+). What happens?
4. Multiply 3 by 14 and subtract 5 from the result.
5. Raise 5 to the power of 4 (hint: use ^ for exponents).
6. Divide 267563 by 1235.

Logical Operators:

7. Use logical operations to get R to agree that "two plus two equals five" is FALSE.
8. Use logical operations to test whether 8 raised to the power 13 is less than $15 \wedge 9$.

Scripts:

9. Use the Rstudio toolbars to open a new script.
10. Use the save button to save it to disk as "using R HW.R".
11. Write some comments at the top of the script indicating the script is your homework.
12. Save the rest of the assignment in your script, and upload that .R file as your homework. Be sure to answer the questions above by pasting what you typed or answering the question asked.

Functions:

13. Use the sqrt() function to calculate the square root of 789.
14. Round 2.456 to the nearest whole number using round().
15. Round 2.456 to two decimal places using round().
16. R has a factorial() function that calculates the factorial function, n! (e.g., 5! = 5*4*3*2*1). Use it to calculate 25!, and take note of the way that R formats the answer.
17. Use factorial() to (try to) calculate 2000!, and note the answer that R gives.

Variables:

18. Create a variable called potato whose value corresponds to the number of potatoes you've eaten in the last week. Or something equally ridiculous.

19. Print out the value of potato by typing the variable name.
20. Do it again using the print() function.
21. Calculate the square root of potato using the sqrt() function.
22. Print out the value of potato again to verify that the value of potato hasn't changed.
23. Reassign the value of potato to potato * 2.
24. Print out the new value of potato to verify that it has changed.
25. Try making a character (string) variable (cheese) and a logical variable (happiness).
26. Try creating a variable with a "missing" value NA.

Vectors:

27. Create a numeric vector with three elements using c().
28. Create a character vector with three elements using c().
29. Create a numeric vector called age whose elements contain the ages of three people you know, where the names of each element correspond to the names of those people.
30. Use "indexing by number" to get R to print out the first element of one of the vectors you created in 27 or 28 or 29.
31. Use negative indices to get R to print out everything except the first element of that vector
32. Use logical indexing to return all the ages of all people in age greater than 20.
33. Use indexing by name to return the age of one of the people whose ages you've stored in age.

Dataframes:

34. For this exercise, we'll use one of the data frames that comes bundled with R, rather than trying to create a new one. The airquality data frame contains 153 cases and 6 variables. You can't actually see it in the workspace because R is storing it in a "hidden" location (sort of). Type airquality at the command line to see what it looks like.
35. Use data(airquality) to load the data into your environment window in studio.
36. Use the $ method to print out the Wind variable in airquality.
37. Print out the third element of the Wind variable.
38. Create a new data frame called aq that includes only the first 10 cases. Hint: typing c(1,2,3,4,5,6,7,8,9,10) is tedious. R allows you to use 1:10 as a shorthand method!
39. Use logical indexing to print out all days (ie. cases) in aq where the Ozone level was higher than 20.
    a.   What did the output do with NA values?
40. Use subset() to do the same thing. Notice the difference in the output.
41. Create a TooWindy variable inside aq, which is a logical variable that is TRUE if Windy is greater than 10, and FALSE otherwise.
42. Delete that variable.

Packages:

43. Install the ggplot2 package.
44. Install the car package.

45. Install the ez package.
46. Load the car library.

Files:

47. Import the Excel file provided on OSF (https://osf.io/jse8h) and print first 6 rows of data