Week One Skills

General principles:

- There is more than one way to do almost anything!
- Google is your friend.
- Infinitely many resources and tutorials.
- Think first, code second.
- Check yourself before you wreck yourself: test your code.
- Reproducibility: What happens if the source data changes?

RStudio:

- I know how to view datasets.
- I know where my environment is.
- I know how to execute code in the console.
- I know how to use the files, plots, packages, and help panes.
- I know how to get helpful information about functions and packages.

Arithmetic:

`+`, `-`, `\`, `*`, `^`, `%%`, `sqrt()`

- I can use R as a calculator

Variable assignment:

`<-`, `class()`

`mutate()`, `rename()` [tidyverse]

- I can create a new variable to store information.
- I can give the variable a reasonable name.
- I can check know the variable type.

Vectors:

`c()`, `rep()`, `seq()`, `:`

- I can creating a list of numbers as efficiently as possible.
- I can do operations on every vector element at once.
- I can combine two vectors.
- I can subset vectors with ‘[]‘

Logical:

`<`, `>`, `==`, `<=`, `>=`, `!`

- I know what a "boolean" is.
- I can subset a vector using logic statements.

Matrices:

`matrix()`, `rowSums()`, `colSums()`, `rbind()`, `cbind()`, `[]`

- I can create a matrix.
- I can find row and column totals of a matrix.
- I can combine two matrices.
- I can select rows, columns, and elements of a matrix.

Categorical Variables:

`factor()`, `levels()`, `summary()`

`case_when()` [tidyverse]

- I can create a factor from a character vector.
- I can find and change the possible categories.
- I can summarize the variable.

Data Frames:

`data.frame()`, `head()`, `tail()`, `summary()`, `str()`, `[]`, `order()`, `sort()`, `subset()`

`tbl_df()`, `arrange()` [tidyverse]

- I know the difference between a matrix and a data frame.
- I can create a data frame.
- I can investigate an unknown data frame.
- I can subset a dataframe.
- I can sort a data frame.

Lists:

`list()`

- I know the difference between a matrix, a data frame, and a list.
- I can create a list.
- I know how to use ‘[]‘ and ‘[[]]‘

Tidyverse and Base R:

- I know the difference between Tidyverse and Base R.
- I have an idea of the advantages and disadvantages of each.

Loops:

`for()`, `while()`, `in`, `apply()`

- I can write a "for" loop using indices.
- I can write a "for" loop from an existing vector or data frame.
- I can write a "while" loop.
- I can use ‘apply()‘ instead of a loop.

Conditionals:

`if()`, `else if()`, `else()`

- I can write useful "if" statements.

Functions:

`function()`, `return()`, `source()`

- I can write my own simple functions.
- I know what a script is.
- I can call scripts from sources to load functions.

Dataset Exploration:

`filter()`, `arrange()`, `select()`, `top_n()`, `desc()`

- I can investigate an unknown data frame.
- I can subset a data frame by column.
- I can subset a data frame by row conditions.
- I can sort a data frame.

Creating variables:

`mutate()`, `summarize()`, `group_by()`,

`mutate_at()`, `summarize_at()`,

`mutate_if()`, `summarize_if()`,

`mutate_all()`, `summarize_all()`

- I can create new variables as needed in a dataset.
- I can find variable attributes, like mean and median, by group(s).

Plotting:

`ggplot()`, `aes()`, `geom_bar()`, `geom_boxplot()`, `geom_histogram()`, `geom_point()`

`ggitle()`, `xlab()`, `ylab()`, `scale_x_continuous()`, `scale_y_continuous()`, `facet()`

- I can make a plot appropriate to my research question
- I can adjust the colors and labels as needed.
- I can make multiple plots across different values of a categorical variable.

Dataset restructuring

`gather()`, `spread()`, `separate()`, `pull()`

- I can split a variable into two separate ones.
- I can convert variable values to a column of categories with corresponding values.
- I can convert a categorical variable to column names.
- I can find the largest and smallest values in a dataset.

Dates and times

`parse_date_time()`, `mdy()`, `hms()`, ...

`day()`, `month()`, `year()`, ...

`force_tz()`, `with_tz()`, ...

`wday()`, `mday()`, ...

- I can create a Datetime object, and I understand how this is different than a character value.
- I can manipulate Datetime objects to find pieces of information or display differently.
- I can edit Datetime objects.
- I understand the integer representation of a Datetime.

Strings

`str_extract`, `str_detect`, `str_replace`, etc.

`str_extract_all`, `str_detect_all`, `str_replace_all`, etc.

`str_trim`, `str_trunc`

`paste`, `str_c`, `print`

- I understand what a string (character) object type is.
- I can use various functions to search and edit vectors of strings.
- I can use regular expressions to match desired parts of strings.
- I can manipulate, combine, and pring strings.

Writing Functions

`<- function()`, `return`

- I can write simple functions.
- I can write "helper functions" to use inside a main function.
- I understand how to choose inputs and outputs for a function.

Functional Programming

`map`, `map_chr`, `map_dbl`, etc.

`map2`, `map2_chr`, `map2_dbl`, etc.

`~function(.x)` notation

`apply`, `sapply`, `lapply`, etc.

- I can use functions in for loops.
- I can use ‘map‘ or ‘apply‘ to apply a function for many values of input.
- I can use special ‘map‘ notation or "headless functions" to apply a function for many values, with multiple inputs.
- I can use ‘map2‘ to apply a function for many values of each input.
- I know what kind of output to expect when I use ‘map‘ or ‘apply‘ type functions.
- I can write simple functions to be used with ‘map‘ or ‘apply‘ type functions.
- (Optional) I can use list-columns with ‘map‘.

Probability Distributions and Tests

`sample`, `sample_n`

`pnorm`, `qnorm`, `dnorm`, `rnorm`,

`p`, `q`, `d`, `r` types for other distributions

`t.test`, `chisq.test`

- I can take samples from a dataset.
- I can take samples from a distribution.
- I can answer probability questions with ‘p‘, ‘q‘, and ‘d‘ functions.
- I can perform t-tests and Chi-square tests in R.