# R CHEAT SHEET (INTRO and DPLYR)

## OPERATORS/SYMBOLS

+ - * / (Math operators; - is also "exclude" in indexing; + is used in ggplot to add new elements)

? (Help operator)

# (Comments operator)

> (Ready prompt; also "greater than")

<- and = (Assignment operators for making objects; = also used to put input w/ arguments)

: (Create a simple sequence)

, (Dimension separator in indexing; argument separator in functions)

! (Negates things--"not that")

$ (Shortcut for indexing a data frame column)

%>% (Pipe for pumping output from one function as input into another)

" " (or ' ') (Marks text)

> < >= <= == (Logic operators--used when filtering)

## CORE CONCEPTS

**ASSIGNMENT** (Creating objects to store data)

name.of.object <- (or =) *values to store*

**FUNCTIONS** (Commands that will do work for you)

function.name(required.input1, optional.input2, …)

**INDEXING** (viewing/modifying contents of objects)

object.name[*value(s) to extract*]

or object.name[*row value(s)*, *column values(s)*]

**TURNING ON PACKAGES**

library(*package_name*) (or use the packages tab)

**SCRIPTS (Text files for saving code for reference/use later)**

**IMPORTING/CHECKING DATA (see useful functions)**

## USEFUL FUNCTIONS (Key arguments)

- log(x, base)
- sqrt(x)
- read.csv(path)

# R CHEAT SHEET (INTRO and DPLYR)

- head(x); tail(x)
- dim(x); nrow(x); ncol(x)
- names(x)
- str(x)
- summary(x)
- c()
- mean(x, trim, na.rm)
- select(data, *column(s) to keep*, …)
- arrange(data, *column(s) to sort by*, …)
- mutate(data, *column(s) to create*, …)
- filter(data, *rule(s) for keeping rows*, …)
- group_by(data, *column(s) to group by*, …)
- summarize(data, *metadata to generate for each group*, …)
- n()

# DPLYR EXERCISES

#1. Make a new data set called *small_surveys* that only has the *species_id*, *sex*, and *weight* columns from the original *surveys* data set.

#2. Make a new data set called *sorted_surveys* that sorts *small_surveys* first by *species_id* in ascending order and then by *weight* in descending order.

#3. Make a new data set called *mutated_surveys* that adds a new column to *sorted_surveys* called *sqrt_weight* that is the square root of the *weight* column (hint, you will need the *sqrt*() function).

#4. Make a new data set called *filtered_surveys* that filters the *mutated_surveys* data set such that we only have data from female animals that weigh less than or equal to 50.

#5. Produce the fully summarized data set (a summary of counts for each species and sex combination) in a single line of code, using pipes and starting from the original *surveys* data set.

DATA SETS TO MAKE FOR THE GGPLOT LESSON

#1. Make a data set called *just_dm* that is only the observations from the species with the id "DM" from the original *surveys* data set.

#2. Make a data set called *stat_summary* that contains the average weight and hindfoot length of each species, as well as a count of the number of observations for each species (using the n() function).

#3. Make a data set called year_summary that contains the **yearly** average weight, hindfoot length, and count data for each species and sex combination (Hint: You only need to change the group_by() part from #2 to do this!).