Exercise: putting it all together

In this exercise, we're going to have you work in pairs to try your hand at some actual data analysis.

Here's the problem: we have four populations of a rare species of fish, characterized by several bright stripes on their bodies. We know that in the populations, we can find individuals with between 1 and 8 stripes, but we don't know how prevalent each stripe number is in each population. What we want to know is: is there any evidence that the average number of stripes vary between the four populations?

Since we don't actually have this data collected, we're going to generate the data as we go. Here's code that will load the three fish population frequencies (don't look at the actual frequencies! Since we can just call weights as individual variable names, we can pretend for the time being that the population frequencies are something to find out, rather than given numbers).

```
fish_freq <- read.csv("Fish_population_frequencies.csv", header=T)
#This is how we can read in a csv (comma separated files). The first
#part specifies the name of the file to read in, and the header
#argument tells R that the first line of this csv file is the column
#labels for the whole data set. Virtually all spreadsheets have a
#"save as csv" option for worksheets, so it's very easy to translate
#your experimental or field data into this format. [1]


pop_1_freq <- fish_freq$pop1
pop_2_freq <- fish_freq$pop2
pop_3_freq <- fish_freq$pop3
pop_4_freq <- fish_freq$pop4
```

Using this data, complete the following exercises; If you don't get them immediately, don't stress about it; the point of this exercise is to get you familiar with writing R code, and get a taste of the expanded list of questions you can ask as you become more proficient.

1 the object fish_freq is what's called a data frame. This is a special data type in R, since so many types of analysis depend on it for input. It's organized like a spreadsheet page, with named columns, which R can treat as variables. We can get information out of it either by variable name, or by index: if you type `fish_freq$sample.var`, you'll get the same result as if you type `fish_freq[,5]`. Similarly, you can get the first entry out of `sample.var` by typing `fish_freq$sample.var[1]`, or `fish_freq[1,5]`

1. First, generate 100 individuals from each population, using the following function (you can call the new population variables whatever you want):

```
fish_stripe_function <-function(number.of.fish,population.freq){
     n.stripes <- length(population.freq)
     fish_population <- sample(x = 1:n.stripes,size = number.of.fish,
          prob = population.freq,replace=T)
     return(fish_population)
     #the return function tells R to stop the function and return the
     #the variable inside the brackets at the output of the function.
     #It's a good idea to get into the habit of using return(), since
     #it makes it easier to tell what the function will be outputting
     #and when the function will terminate and send output.
}
```

2. Using help files and internet searches, look up how to run a t-test[2] on your data; Create a table of comparisons between all the different populations, and try and determine which, if any, of the sub-populations might actually be distinct from each other.

3. Create summaries of the data for each sub-population (I suggest histograms for this, but you might also try using the `summary()` command, or the `boxplot()` command); Does the data support your previous conclusions about the differences between populations?

2 A t-test is a statistical test to see whether the means (arithmetic averages) of two populations differ by a statistically significant amount