**Exercise 6 – converting wide format to long format**

The goal of this assignment is to create your first useful R script. Your script should start with the raw data from AMT (raw.data.from.AMT.csv), which is in wide format, and convert it to long format (such that it looks like results.long.format.no.items.csv).

Now, I know that there are several packages out there that will do this in a relatively automated way (reshape2, tidyr). The point here is to practice writing an R script that will do the conversation in a way that mimics doing it by (i.e., using copy and paste, and filling in labels). I will help you by telling you all of the functions that I used in my version of this script. Most likely you will use them too (though, you do not need to do use them all if you don’t need them for your version of the script). You should, of course, feel free to use google to help you with this… but don’t google for the solution directly. Google smaller questions that you ask yourself as you try to figure out how to do this.

Remember, the first step of any script is to write down the steps you need to achieve in plain English. Then you can work through those steps, converting them to R code.

When you are finished, submit your script to me. I will run it using raw.data.from.ATM.csv, and look to see if it creates results.long.format.no.items.csv.

Functions that may be helpful

[] – bracket notation for indexing data sets

1:10 – the colon operator creates a sequence of integers (here from 1 to 10, as an example)

read.csv() – reads in csv files.

sub() – replaces character strings with other character strings

subset() – creates a subset from a data set based on a logical statement

nrow() – counts the number of rows in a matrix or data frame

ncol() – counts the number of columns in a matrix or data frame

order() – useful for re-ordering a dataset according to a column (you can google this)

as.numeric() – treats non-numeric elements as numeric

as.character() – treats non-character data as character strings

as.vector() – treats a non-vector as a vector; you can flatten a matrix into a vector using this.

as.matrix() – treats a two-dimensional object as a matrix

cbind() – combines two matrices along a column edge

rbind() – combines two matrices along a row edge

t() – transposes a matrix

rep() – repeats an element, or a sequence of elements, a certain number of times

data.frame() – creates a data frame from separate elements (vectors, matrices)

write.csv() – writes a csv file