Creativity2.r: Explanation

Goals of code:

- Read a space delimited file
- Do a randomization test

Reading a space delimited file:
`creativity <- read.table('creativity.txt', as.is=T, header=T)`
Space delimited files are read by read.table(). In fact, so can comma delited files, but read.csv() is simpler for those.

The first argument is the name of the file. Note the name includes the file type (also called the file extension). The second argument is optional: as.is=T turns off a default behaviour that can lead to confusion (more some weeks from now when we discuss factors). So is the third argument, header=T, but you want to include that for any 401 data file (and probably for all of your data files). header=T indicates that the first line in the file is to be interpreted as a header line containing the variable names.

The output of this function is a data frame containing the contents of the data file. The `<-` assigns this to a variable, in this case creativity.

If you look at the resulting data frame and see variables called V1, V2, ..., you forgot header=T. R made up variable names (since it wasn't told them) and the contents of the variable will be a mix of a name and the real data. That's garbage. Rerun read.table() with header=T.

Randomization p-value: The code from `response <- creativity$score` to `p.value`
There is no default function to compute the randomization test p-value. This code uses R's programming commands to do that. In this example, `creativity$score` contains the response variable; `creativity$treatment` contains the grouping variable. These are copied into two new variables, response and group, which are the variables used elsewhere in the code. To use for a new problem, change the variables on the right-hand side of the first two commands, then execute all lines down to and including the p.value line.

The two-sided p-value is printed after you execute the last line.

A short description of what the code does:
save the number of observations and the unique names of the groups
create a vector of logical values with TRUE when that observation came from the first group
`obsdiff` is the difference in means between the first group and any other observations
the core of the code is a loop, executed once for each random assignment of labels to observations
`sample(group)` permutes the group labels
we then identify which randomly permuted observations were in the first group, compute the mean difference, and save it in vector
the two-sided p-value is then the number of more extreme random differences, with +1 to include the observed sample.

I am happy to provide more explanation if you want to know the details.

Note: Those of you familiar with computer programming will realize this code could be easily converted to a user-defined function. I agree. We'll define our own functions later.