Introduction to repetitive tasks in R

## When would I need to perform repetitive tasks?

Loading multiple data files, repeating an operation (for example descriptive statistics) in different groups or by different conditions, or generating a sequence.

## What are the different ways you can repeat tasks in R?

Loops, vectorized (lapply and friends), functions.

## When should you use each case?

Loops are useful for modifying an object repeatedly in place, recursive functions, and running until a condition is met [while loops](http://adv-r.had.co.nz/Functionals.html#functionals-not)

However, R is vectorized which means most looping, such as going through rows of a database, can be performed by applying a function to a column by itself with no further work. lapply() and its related functions (tapply(), sapply()) let you apply functions in groups.

Writing your own function can help with complex loops or be used in lapply(). Loops and vectorized operations are similar at small-moderate sized datasets, but in larger datasets vectorizations should be faster. Sequences require a loop. Otherwise, you can use [lapply, tapply, or sapply](https://rpubs.com/markpayne/132503)

## Where can I find more code and examples like you have in this workshop?

The content in this workshop was adapted from the Software Carpentry lessons for [gapminder](https://swcarpentry.github.io/r-novice-gapminder/) and [inflammation datasets](http://swcarpentry.github.io/r-novice-inflammation/) and from Hadley Wickham’s open licensed book [Advanced R](http://adv-r.had.co.nz).