Enumerable#cycle

Using cycle would allow you to call the block for each element repeatedly for as many times as you want or forever if the amount of cycles are not specified. A basic example of this is:

numbers= [“dog”, “cat”, “me”, “you”]

numbers.cycle{ |x| puts x } # prints (“dog”, “cat”, “me”, “you”) over infinitely.

You can specify how many times this array can be called like this.

numbers= [“dog”, “cat”, “me”, “you”]

numbers.cycle(5) { |x| puts x } # prints (“dog”, “cat”, “me”, “you”) #exactly 5 times and then stops.

This would have been super useful for the formula I used for my accountability group. I might go back and redesign it for this. In my original formula I was able to take a list of 24 names and retrieve outputs of 6 groups with 4 random individuals. If I use the .cycle enumerable, I will be able to call 3 different sets of 6 groups with 4 different people in each group.