Assignment 2: Coding Basics

Michael Gaffney

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on coding basics.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your first and last name into the file name (e.g., "FirstLast_A02_CodingBasics.Rmd") prior to submission.

Basics Day 1

1. Generate a sequence of numbers from one to 100, increasing by fours. Assign this sequence a name.

2. Compute the mean and median of this sequence.

3. Ask R to determine whether the mean is greater than the median.

4. Insert comments in your code to describe what you are doing.

```r
#1.create sequence of numbers from 1 to 100, skipping by four
numbers <- seq(1,100,4)

#2. set mean and median of the sequence to variables
a <- mean(numbers)
b <- median(numbers)
#3. evaluate variables with a boolean
a > b
```

```
## [1] FALSE
```

Basics Day 2

5. Create a series of vectors, each with four components, consisting of (a) names of students, (b) test scores out of a total 100 points, and (c) whether or not they have passed the test (TRUE or FALSE) with a passing grade of 50.

6. Label each vector with a comment on what type of vector it is.

7. Combine each of the vectors into a data frame. Assign the data frame an informative name.

8. Label the columns of your data frame with informative titles.

```r
#character vector of student names
students <- c("Matthew", "Mark", "Luke", "John")
#numerical vector of scores
```

```
scores <- c(95, 76, 49, 85)
#logical vector of pass fail
passed <- c(TRUE, TRUE, FALSE, TRUE)
#create grades dataframe
grades <- data.frame("name" = students, "score" = scores, "pass/fail" = passed)
print(grades)
```

```
##      name score pass.fail
## 1 Matthew   95      TRUE
## 2    Mark   76      TRUE
## 3    Luke   49     FALSE
## 4    John   85      TRUE
```

9. QUESTION: How is this data frame different from a matrix?

   Answer: They're similiar, but a matrix can only store one type of data (like numeric, for example), where a dataframe can store multiple forms of data.

10. Create a function with an if/else statement. Your function should determine whether a test score is a passing grade of 50 or above (TRUE or FALSE). You will need to choose either the `if` and `else` statements or the `ifelse` statement. Hint: Use `print`, not `return`. The name of your function should be informative.

11. Apply your function to the vector with test scores that you created in number 5.

```
passing <- function(x){
  y <- ifelse(x>=50,TRUE,FALSE)
  print(y)
}

passing(grades$score)
```

```
## [1]  TRUE  TRUE FALSE  TRUE
```

12. QUESTION: Which option of `if` and `else` vs. `ifelse` worked? Why?

    Answer: I used ifelse becuase the outcome I needed for the scores was binary–if it's above 50, then it's true, and if it's below, it's false. The ifelse function is the simplest and most efficient way to create this output.