Assignment 2: Coding Basics

Camber Vincent

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A02_CodingBasics.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.
6. After Knitting, submit the completed exercise (PDF file) to Canvas.

Basics, Part 1

1. Generate a sequence of numbers from one to 55, increasing by fives. 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.

```
#1.
fives_sequence <- seq(1,55,5) # Generate a sequence of numbers from 1 to 55 by 5

#2.
mean_fives <- mean(fives_sequence) # Compute the mean of the sequence
median_fives <- median(fives_sequence) # Compute the median of the sequence

#3.
mean_fives > median_fives # Determine if the mean is greater than the median
```

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

Basics, Part 2

5. Create three vectors, each with four components, consisting of (a) student names, (b) test scores, and (c) whether they are on scholarship or not (TRUE or FALSE).

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.

```
student_names <- c("Anthony","Bethany","Cameron","Delila") # Character vector
test_scores <- c(1400,1230,1600,1420) # Numeric vector
scholarship <- c(TRUE,FALSE,TRUE,FALSE) # Logical vector

student_profiles <- data.frame(student_names,test_scores,scholarship) # Create data frame
colnames(student_profiles) <- c("Students","Test Score","On Scholarship?") # Informative column titles
```

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

   Answer: This data frame is primarily different from a matrix because a matrix must be made entirely of the same data type, i.e. all numeric, while a data frame can contain different types of data in different columns.

10. Create a function with one input. In this function, use `if...else` to evaluate the value of the input: if it is greater than 50, print the word "Pass"; otherwise print the word "Fail".
11. Create a second function that does the exact same thing as the previous one but uses `ifelse()` instead if `if...else`.
12. Run both functions using the value 52.5 as the input
13. Run both functions using the **vector** of student test scores you created as the input. (Only one will work properly...)

```
#10. Create a function using if...else
function_gt50 <- function(x){
  if(x>50){"PASS"}
  else{"FAIL"}
}

#11. Create a function using ifelse()
function_gt50_2 <- function(x){
  ifelse(x>50,"PASS","FAIL")
}

#12a. Run the first function with the value 52.5
function_gt50(52.5)
```

```
## [1] "PASS"
```

```
#12b. Run the second function with the value 52.5
function_gt50_2(52.5)
```

```
## [1] "PASS"
```

```
#13a. Run the first function with the vector of test scores
# function_gt50(test_scores)
# Commented out in order to properly knit

#13b. Run the second function with the vector of test scores
function_gt50_2(test_scores)
```

```
## [1] "PASS" "PASS" "PASS" "PASS"
```

14. QUESTION: Which option of `if...else` vs. `ifelse` worked? Why? (Hint: search the web for "R vectorization")

    Answer: The second version of the code using the ifelse function worked. The difference between the functions is because of vectorization. Some functions in R are vectorized, meaning they can operate on an entire vector of data at once rather than requiring only one data entry The ifelse function is vectorized, so if a vector of data is input it can evaluate the logical statement for each value in the vector and return a vector of results. The if…else function is not vectorized and expects a single logical value, thus it does not work.

NOTE Before knitting, you'll need to comment out the call to the function in Q13 that does not work. (A document can't knit if the code it contains causes an error!)