Assignment 2: Coding Basics

Kendall Fitzgerald

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.
vector1 <- c(seq(from=1,to=55,by=5))
vector1 # I am using seq() function to increase from 1 to 55 by five's and have named it 'vector1'
```

```
##  [1]  1  6 11 16 21 26 31 36 41 46 51
```

```
#2.
mean(vector1) # Calculating mean of 'vector1' using RStudio's built-in software
```

```
## [1] 26
```

```
median(vector1) #calculating median of 'vector1'
```

```
## [1] 26
```

```r
#3.
mean(vector1)>median(vector1)
```

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

```r
# said that mean was greater than median of 'vector1' which R determines as 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.

```r
#creating three vectors of equal length with required information
names <- c('Sarah', 'Thomas', 'Mary', 'John')
testscores <- c(82, 93, 46, 78)
scholarship <- c(TRUE, FALSE, FALSE, TRUE)

#Combining the vectors into a dataframe and naming dataframe "StudentStats"
StudentStats <- data.frame(names,testscores,scholarship)

#Setting the column names
names(StudentStats) <- c("Name","Score","Scholarship")
```

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

   Answer: This data frame is different from a matrix because matrices can only contain one class of data (i.e. numbers, names, etc.) while data frames, including this one, can contain different classes of data.

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...)

```r
#10. Create a function using if...else
x=10
if(x > 50){
print("Pass")
} else {
print("Fail")
}
```

```
## [1] "Fail"
```

```
#11. Create a function using ifelse()

ifelse(x>50,"Pass","Fail")
```

```
## [1] "Fail"
```

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

x=52.5
if(x > 50) print("Pass") else print("Fail")
```

```
## [1] "Pass"
```

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

x=52.5
ifelse(x>50,"Pass","Fail")
```

```
## [1] "Pass"
```

```
#13a. Run the first function with the vector of test scores

# x=testscores
# if(x > 50) print("Pass") else print("Fail") #commenting out this function so that it can be knitted

#13b. Run the second function with the vector of test scores

x=testscores
ifelse(x>50,"Pass","Fail")
```

```
## [1] "Pass" "Pass" "Fail" "Pass"
```

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

    Answer: The second 'ifelse' function worked because it is a more complex vectorized function that can cycle through data that was already made as a vector. The 'if'... 'else' function did not work because it is a more rudimentary loop function that isn't as commonly used in R as in other coding languages. In short, the 'if' function is not vectorized whereas 'ifelse' is.

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!)