Assignment 2: Coding Basics

Camila Rodriguez

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.

```
seq(1,55,5) #(from, to, by)
```

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

```
fiftyfive_spacedfive <- seq(1,55,5)
fiftyfive_spacedfive
```

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

2. Compute the mean and median of this sequence.

```
mean(fiftyfive_spacedfive)
```

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

```
median(fiftyfive_spacedfive)
```

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

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

```
mean(fiftyfive_spacedfive)>median(fiftyfive_spacedfive)
```

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

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

```
#1. Sequence is (from, to,by) to show numbers from 1 through 55 spaced by 5

#2. Mean of fiftyfive_spaced_five is calculated by mean function; median is calculatd by median functio

#3. I use the > sign for R to respond if mean is greater than the mode
```

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).

```
studentnames <-c("Victoriah","Alice","Lord", "Dominic") #Character
grades <-c (82, 80,93,98) #numeric
scholarship <- c(TRUE, FALSE, FALSE, TRUE) #logical
```

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.

```
studentsummary<- data.frame(studentnames, grades, scholarship)
```

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

```
names(studentsummary)<- c("Name", "Grade","Scholarship");
```

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

   Answer:In a matrix you have to define how many rows and columns you want as an output given the data you have. A matrix is homogenous, unlike data frames which can have heterogeneous data. Visualization is also better for data frames compared to matrices.

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

larger_than_fifty <- function(x) {if(x>50){
  print("Pass")
} else {print("Fail")}

}


#11. Create a function using ifelse()

larger_than_fifty_version2 <- function(x){
  ifelse(x>50, "Pass", "Fail") #(condition, then, else)

}

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

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

```r
#12b. Run the second function with the value 52.5
larger_than_fifty_version2(52.5)
```

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

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

#larger_than_fifty(grades)

#13b. Run the second function with the vector of test scores
larger_than_fifty_version2(grades)
```

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

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

    Answer: Since most of R funcitons are vectorized, "ifelse" is used for vector based calculations. "If...else" is not vectorized, making it more prone to errors while running. Operations under "ifelse" occur on an element by element basis, allowing for smoother calculations compared to "if... else" which may cause more errors.

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!)