

Assignment 2: Coding Basics

Lauren Shohan

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.
sequence1to55_by5 <- seq(1,55,5) #from, to, by
sequence1to55_by5
```

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

```
#2.
mean_sequence55 <- mean(sequence1to55_by5) #mean of the sequence
median_sequence55 <- median(sequence1to55_by5) #median of the sequence

#3.

mean_sequence55 > median_sequence55 #is the mean 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.

```
#5.
student_names <- c("Lauren", "Nicole", "Sophie", "Sam") #Character vector
test_scores <- c(99,92,78,85) #Numeric vector
scholarship <- c(TRUE,FALSE,TRUE,FALSE) #Logical vector

dataframe_students_scores_scholarship <- data.frame(student_names,test_scores,scholarship)

names(dataframe_students_scores_scholarship) <- c("Student Names","Test Scores","Scholarship")
```

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

Answer: This data frame is more general than a matrix. This data frame can have different modes for the columns like numeric, character, logical while a matrix contains elements of the same type.

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 of `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

function1_if_and_else <- function(x) {
  if(x > 50){
    print("Pass")
  }
  else{
    print("Fail")
  }
}

#11. Create a function using ifelse()

function2_ifelse <- function(x) {
  ifelse(x > 50, "Pass", "Fail")
}

#12a. Run the first function with the value 52.5
function1_if_and_else(52.5)
```

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

```
#12b. Run the second function with the value 52.5  
function2_ifelse(52.5)
```

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

```
#13a. Run the first function with the vector of test scores  
#function1_if_and_else(test_scores) #error  
  
#13b. Run the second function with the vector of test scores  
function2_ifelse(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 `ifelse()` second function worked while the first one with `if` and `else` did not work and produced an error. The `ifelse()` worked because this function is made to handle and run vectors whilst `if ... else` cannot support vectors.

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!)