Assignment 2: Coding Basics

Darpan Barua

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.

sequence <- 1 + (0:10)*5

sequence

2. Compute the mean and median of this sequence.

mean_sequence <- mean(sequence)

median_sequence <- median(sequence)

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

mean_greater_than_median <- mean_sequence > median_sequence

mean_greater_than_median

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

```
#1. Each integers from 0:10 is multiplied by 5. We add 1 to the product since that's the first number w

#2. Here we use the basic functions of mean and median to find the relevant data. We use the variables

#3. Here we note a context as "mean_greater_than_median" and denote it as such that mean_sequence is gr
```

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).

student_names <- c("Terry", "Tim", "Nino", "Bethany") #character vectors

print(student_names)

test_scores <- c(85, 92, 78, 88) #numeric vectors

print(test_scores)

on_scholarship <- c(TRUE, FALSE, TRUE, FALSE) #logical vectors

print(on_scholarship)

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

#Please check above

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

student_data <- data.frame(

```
Name = student_names, #Name of the student
Score = test_scores, #Score obtained by the student on the test
Scholarship = on_scholarship #Is the student on scholarship?
```

)

print(student_data)

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

#Done. Please check above!

```
#5. We create variables and assign the set of components that define those variables.
#6. We simply set comments next to each line of code.
#7. We use the data.frame function to create a data frame.
#8. Comment added next to the data frame elements.
```

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

   Answer: Data frame has different types of vector data while a matrix only has one 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".

check_pass_if<-function(score) {

```
if (score>50) {

  print("Pass")
```

} else {

```
  print("Fail")
```

}

}

11. Create a second function that does the exact same thing as the previous one but uses `ifelse()` instead if `if...else`.

check_pass_ifelse<-function(score) {

```
  print(ifelse(score>50, "Pass", "Fail"))
```

}

check_pass_ifelse(test_scores)

12. Run both functions using the value 52.5 as the input

check_pass_if(52.5)

check_pass_ifelse(52.5)

13. Run both functions using the **vector** of student test scores you created as the input. (Only one will work properly...)

check_pass_if(test_scores) #does not work

check_pass_ifelse(test_scores) #works

```
#10. Create a function using if...else - this was a basic function which I denoted using variable "chec

#11. Create a function using ifelse() - this was a shorter function which I denoted using variable "che

#12a. Run the first function with the value 52.5 - Inserted 52.5 in place of 'test_scores' in the 'chec

#12b. Run the second function with the value 52.5  - Inserted 52.5 in place of 'test_scores' in the 'ch

#13a. Run the first function with the vector of test scores - Inserted 'test_scores' in the 'check_pass

#13b. Run the second function with the vector of test scores - Inserted 'test_scores' in the 'check_pas
```

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

    Answer:'if'...'else' is designed for singular value inputs (scalar) and it only works with one number at a time. 'ifelse' is designed for entire vectors (vectorized) and therefore applying the condition element-wise to each score in the vector.

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!)