Assignment 2: Coding Basics

Jazmine Pritchett

OVERVIEW

This exercise accompanies the lessons 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 Sakai.

Basics, Part 1

1. Generate a sequence of numbers from one to 30, increasing by threes. 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.
seq(1,30,3) #from,to,by
```

```
##  [1]  1  4  7 10 13 16 19 22 25 28
```

```
three_sequence <- seq(1,30,3)

#Using "seq" function to list every three numbers of a
#sequence of numbers from 1 to 30. Named this sequence
#"three_sequence"by assigning the sequence value to the object.

#2.
mean(three_sequence)
```

```
## [1] 14.5
```

```r
median(three_sequence)
```

```
## [1] 14.5
```

```r
#Using the object above that has the sequence, plug in to mean
#function and then median function.

#3.
14.5 > 14.5
```

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

```r
#Conditional statement to compare if the mean is greater than the
#median. With true being yes and false being no.
```

Basics, Part 2

5. Create a series of vectors, each with four components, consisting of (a) names of students, (b) test scores out of a total 100 points, and (c) whether or not they have passed the test (TRUE or FALSE) with a passing grade of 50.
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
#This vector contains characters of student's names.
students <-c("Clarke", "Gabriel", "Dylan", "Ria")

#This numerical vector is of test scores of the 4 students.
test_scores <- c(72,46,98,51)

#Test scores are assigned to students. Logical vector is use to
#assign scores greater than or equal to 50 as TRUE and those not as FALSE.
test_scores <- c(Clarke= 72, Gabriel=98,Dylan=46,Ria=51)
test_scores>=50
```

```
##  Clarke Gabriel   Dylan     Ria
##    TRUE    TRUE   FALSE    TRUE
```

```r
Student <-c("Clarke","Gabriel","Dylan","Ria")

Score <-c(72,98,46,51)

Passed <-test_scores>=50

StudentTestScores <-data.frame(Score, Passed)

print(StudentTestScores)
```

```
##         Score Passed
## Clarke     72   TRUE
## Gabriel    98   TRUE
## Dylan      46  FALSE
## Ria        51   TRUE
```

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

   Answer: A matrix has columns of the same mode and the same length.It is more general having different modes of characters of student names, student scores, and a logical pass or fail value.

10. Create a function with an if/else statement. Your function should take a **vector** of test scores and print (not return) whether a given test score is a passing grade of 50 or above (TRUE or FALSE). You will need to choose either the `if` and `else` statements or the `ifelse` statement.

11. Apply your function to the vector with test scores that you created in number 5.

```
ifelse(test_scores>=50,TRUE,test_scores)
```

```
##  Clarke Gabriel   Dylan     Ria
##       1       1      46       1
```

12. QUESTION: Which option of `if` and `else` vs. `ifelse` worked? Why?

    Answer:'if' and 'else' returned an error, and 'ifelse' worked. It seems that 'ifelse' checks all elements of the vector to avoid the error of 'if' checking one element at a time and generating the error. If test scores were equal or greater than 50, if assigned it 1 as TRUE, if not then score was generated.