Assignment 2: Coding Basics

Masha Edmondson

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on coding basics.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Salk_A02_CodingBasics.Rmd") prior to submission.

The completed exercise is due on Tuesday, January 21 at 1:00 pm.

Basics Day 1

1. Generate a sequence of numbers from one to 100, increasing by fours. 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.
   my_sequence <- seq(0, 100, by = 4)
#naming the sequence "my sequence" and using the sequence function to create a sequence of numbers from

#2.
   mean(my_sequence)  #finding the mean of the sequence through the mean function
```

```
## [1] 50
```

```
   median(my_sequence) #finding the median of the sequence through the median function
```

```
## [1] 50
```

```
#3.
   mean(my_sequence) > median(my_sequence)
```

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

```
   #asking a logical statement if the mean of the sequence is greater than the median of the sequence
```

Basics Day 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.

```
students <- c("Abby Brown", "Joe Fields", "Eddie Cook", "Fran Johnson") #creating a character vector wi

test_scores <- c(100, 82, 75, 43) #creating a numeric vector with each student's test scores

passed <- test_scores >= 50 #creating a logical vector showing which students passed the exam with a sc

test_results <- data.frame(students, test_scores, passed) #combining vectors into a dataframe and namin

names(test_results) <- c("Student","Test Score","Passed Exam"); View(test_results) #labeling the column
```

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

   Answer: In a data frame, the columns can contain different types of data (numeric, logical, and character statements), but in a matrix all the elements are the same type of data which are usually numbers. Data frames can also combine features of matrices and lists, and the items of the list serve as the columns of the data frame.

10. Create a function with an if/else statement. Your function should determine whether a 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. Hint: Use `print`, not `return`. The name of your function should be informative.
11. Apply your function to the vector with test scores that you created in number 5.

```
#test_scores <- c(100, 82, 75, 43)
#for (value in test_scores){print(value)}
#for (value in test_scores){
  #if (test_scores >= 50){print("Passed Exam")} else {print("Failed Exam")}
#tried to create and "if" and "else" statement function for the test score results, which did not end u

test_results2 <- ifelse(test_scores >= 50, "passed exam", "failed exam") #created "ifelse" function to
test_results2 #named the outcome of ifelse function
```

```
## [1] "passed exam" "passed exam" "passed exam" "failed exam"
```

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

    Answer: When I tried to use the "if" and "else" statement function the code failed to run because the condition has a length that was greater than 1 and only the first element was used- the first test score. This failed because we were trying to use a vector on the function when the if, else statement would only give you a response to one element and not each of the test scores. The "ifelse" function worked because the "ifsele" function returns a value with the same shape as test, an object which can be coerced to logical mode, which is filled with elements selected from either yes (retun value for true elements) or no (return values for false elements) depending on whether the element of test is true or false. This function was able to take our vector data with four elements and produce a logical statement for each of the four elements to determine which students passed or failed the exam.