Assignment 2: Coding Basics

Andrew Friedman-Herring

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 first and last name into the file name (e.g., "FirstLast_A02_CodingBasics.Rmd") prior to submission.

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.
byfour<-seq(1,100,4) #this line of code makes the sequence and gives it the name "byfour"
byfour
```

```
##  [1]  1  5  9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69 73 77 81 85 89 93 97
```

```
#2.
mean(byfour)
```

```
## [1] 49
```

```
median(byfour) #these functions calcualte the mean and median of the sequence
```

```
## [1] 49
```

```
#3.
isTRUE(mean(byfour)>median(byfour))
```

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

```
#this line evaluates the logical statement to see if the mean is larger than the
#median of the saved 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.

```
student_names<-c("Mark", "Jesse", "Michelle", "Kyler") #character vector
test_scores<- c(66, 72, 100, 25) #numberic vector
passed_test_YN<- c(TRUE, TRUE, TRUE, FALSE) #logical vector
```

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.

```
examresults<- data.frame(student_names,test_scores,passed_test_YN)
#because the vectors I am combining already have informative names,
#the columns of the data frame will be also have the same names
examresults
```

```
##   student_names test_scores passed_test_YN
## 1          Mark          66           TRUE
## 2         Jesse          72           TRUE
## 3      Michelle         100           TRUE
## 4         Kyler          25          FALSE
```

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

   Answer: A matrix only holds one kind of data, whereas in a data frame, each column can be of a different vector class

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.

```
exam_pass_YN <- function(x){
  ifelse(x>=50, TRUE, FALSE)
}
```

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

```
exam_pass_YN(test_scores)
```

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

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

    Answer: Both would work, but using `ifelse` is a more efficient way to code the function using less space