Assignment 2: Coding Basics

Griffin Bird

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 Day 1

1. Generate a sequence of numbers from one to 100, increasing by fours. Assign this sequence a name.

```
seq_by4 <- seq(0, 100, by = 4)

head(seq_by4)
```

```
## [1]  0  4  8 12 16 20
```

2. Compute the mean and median of this sequence.

```
seq_mean <- mean(seq_by4)
seq_median <- median(seq_by4)
```

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

```
seq_mean > seq_median
```

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

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

```
#1. I used the sequence function to generate numbers from 1 to 100, and used
#"by" to indicate that the sequence should increase by increments of 4

#2. I calculated the mean and median of the data set using Rs
#built in functions, then assigned those values to variables

#3. I used a boolean expression to get R to tell me whether the mean was greater than
# the median, R will return true or false, and it turns out the mean is not greater than the median.
```

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.

```
students <- c('John', 'Jane', 'Jill', 'Joseph', 'Jacqueline', 'Josephine')
#Character vectors

scores <- c(4, 49, 89, 99, 35, 80)
#Numeric vectors

pass <- (scores>50)
#Logical vectors

print(scores)
```

```
## [1]  4 49 89 99 35 80
```

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

```
tests.df <- data.frame(pass, scores, students)
```

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

```
colnames(tests.df) <- c('Passed', 'Test Score', 'Student Name')
```

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

   Answer: The data frame contains different types of vectors, a matrix can only contain one type of vector

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.

```
did.they.pass <- function(x) { ifelse(x >50, "They passed", "They failed")}
```

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

```
did.they.pass(scores)
```

```
## [1] "They failed" "They failed" "They passed" "They passed" "They failed"
## [6] "They passed"
```

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

    Answer: 'ifelse' worked because it is a vectorized version of 'if', 'if' works with things that have a length of 1 but 'ifelse' can work on an entire vector