Assignment 2: Coding Basics

Nadia Barbo

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.
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.  creating sequence of numbers from 1 to 100 increasing
# by a factor of 4
seq100 <- seq(1, 100, 4)

# 2.
mean(seq100)  #finding the mean of seq100
```

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

```
median(seq100)  #finding the median of seq100
```

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

```
# 3. determining if the mean of seq100 is greater than the
# median of seq100 (TRUE or FALSE)
mean(seq100) > median(seq100)
```

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

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.

```
#5.
studentnames <- c("Nadia", "Karina", "Abby", "Emma") #creating vector of names of students
testscores <- c(12, 87, 53, 100) #creating vector of test scores
passedtest <- c(FALSE, TRUE, TRUE, TRUE) #creating vector of students passing test or not

#6.
typeof(studentnames) #studentnames is a character (chr) vector
```

```
## [1] "character"
```

```
typeof(testscores) #testscores is a double/numeric (num) vector
```

```
## [1] "double"
```

```
typeof(passedtest)#passed test is a logical (logi) vector
```

```
## [1] "logical"
```

```
#7. and 8.
#combining previously created vectors into data frame and naming columns of the data frame
testresults.df <- data.frame("Student.Names" = studentnames, "Test.Scores" = testscores,
                             "Passed.Test" = passedtest)
```

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

   Answer: A data frame is different from a matrix because a data frame can be composed of columns/vectors that are different modes from one another (but they all need to be the same length), while in a matrix, all of the columns need to be the same mode and same length.

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.

```
#10.
#creating function where you input a vector and if a value in that vector is >=50 it will
#print TRUE in it's place and if it is <50 it will print FALSE
passed <- function(grade.vector){
    result <- ifelse(grade.vector>=50, TRUE, FALSE)
    print(result)
}
```

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

```
#11.
#running the function I just created on testscores vector
passed(testscores)
```

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

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

    Answer: The "ifelse" option worked because it is able to check multiple values from a vector at one time, allowing us to input a vector and print the outputs for each value in that vector. The "if" statement can only check one element in a vector at a time, so it cannot evaluate the entire vector.