Assignment 2: Coding Basics

David Liddle

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.
#Generating a sequence of numbers from 1-30, increasing by 3.
x<-seq(1, 30, by = 3)
#Show Sequence
x
```

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

```
#2.
#Calculating mean
x_mean<-mean(x)
#Show mean
x_mean
```

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

```
#Calculating median
x_median<-median(x)
#Show median
x_median
```

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

```
#3.
#Is x_mean > m_median?
x_mean>x_median
```

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

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.

```
#Create a list of student information
  #The student names are a character vector.
  student_names = c("Bethany","Christina","Jaxon","Kyler")
  #The test scores are a numeric vector.
  test_scores = c(48, 92, 88, 95)
  #The passing test scores are a logical vector.
  passed_test = c(FALSE, TRUE, TRUE, TRUE)

#Create a data frame from the vectors
student_data <- list(student_names, test_scores,passed_test)

# Print the student data
print(student_data)
```

```
## [[1]]
## [1] "Bethany"   "Christina" "Jaxon"     "Kyler"
##
## [[2]]
## [1] 48 92 88 95
##
## [[3]]
## [1] FALSE  TRUE  TRUE  TRUE
```

```
# I utilized chatGPT with creating the student information, grades, and vectors.
```

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

Answer: A data frame differs from a matrix because in a matrix, the data must all be the same data type (such as numerical, character, or logical) but a data frame can uitilize multiple data types.

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.

```r
#Create a function with if/else or ifelse command
passing_check<-function(test_scores){
  ifelse(test_scores>=50,"Pass","Fail")}

  #Print test scores
  passing_check(test_scores)
```

```
## [1] "Fail" "Pass" "Pass" "Pass"
```

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

Answer: The ‘ifelse’ command worked while the ‘if’ and ‘else command did not. When I tried to create a function if the if/else method I received this error: Error in if (test_scores >= 50) { : the condition has length > 1. The ’ifelse’ function worked because it is a vectorized form of the ‘if/else command.’