Assignment 2: Coding Basics

Ruiqing Li

OVERVIEW

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

Directions

1. Rename this file `<Li>_A02_CodingBasics.Rmd` (replacing `<Ruiqing>` 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.
first_sequence <- seq(1, 100, 4) #Naming the sequence first_sequence

#2.
mean(first_sequence) #Used sequence's assigned name first_sequence to get the mean
```

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

```
median(first_sequence) #Used sequence's assigned name first_sequence to get the median
```

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

```
#3.
mean(first_sequence) > median(first_sequence) #Using Conditional Statement to compare mean of sequence
```

```
## [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.

```
Names <- c("Lisa","Cassie","Natalie","David") # Character Vectors
Names
```

```
## [1] "Lisa"    "Cassie"  "Natalie" "David"
```

```
TestScores <- c(23,48,92,99) #Numeric Vectors
TestScores
```

```
## [1] 23 48 92 99
```

```
PassTestorNot <-c(FALSE,FALSE,TRUE,TRUE) #Logical Vectors

Grades <-data.frame(Names, TestScores, PassTestorNot)
Grades
```

```
##     Names TestScores PassTestorNot
## 1    Lisa         23         FALSE
## 2  Cassie         48         FALSE
## 3 Natalie         92          TRUE
## 4   David         99          TRUE
```

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

   Answer:Data frame can consist of multiple different classes of data instead of one.

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.

```
#If and Else Function
Test_Score <- function(x){
  pass <- if (x>=50){
    TRUE
  }  else {
    FALSE
  }
  return(pass)
}
```

```
#IfElse Function
Test_Score2 <- function(x){
  pass <- ifelse (x>=50,TRUE,FALSE)
  return(pass)
}
```

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

#If and Else Function ifandelse<- Test_Score (TestScores) ifandelse **Error in if (x >= 50) { : the condition has length > 1**

```
#IfElse Function
ifelse<- Test_Score2 (TestScores)
ifelse
```

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

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

    Answer:`ifelse` function wroked. The vector TestScores have 4 different scores, `ifelse` function can run multiple numbers at a time where `if` and `else` function can only run a single number at a time.