Assignment 2: Coding Basics

Suad Muradov

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.

```
seq_by_4<-seq(1,100,4)
```

2. Compute the mean and median of this sequence.

```
mean(seq_by_4)
```

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

```
median(seq_by_4)
```

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

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

```
mean(seq_by_4)>median(seq_by_4)
```

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

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

```
#1. Assign sequence from 1 to 100 by 4 a name

#2. Find the mean and median valeus of the sequence

#3. Compare the values of median and mean
```

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.

```
Students<-c("Suad","Nijat","Aynur","Rashad") #character
Score<-c(80,100,39,50) #numeric
Pass<-c(TRUE,TRUE,FALSE,TRUE) #logical

Test<-data.frame("student"=Students, "score"=Score, "passed"=Pass)
```

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

   Answer: A matrix's columns must all have the same data type and length (for example, numeric or character). An advantage of using a data frame instead of a matrix is that it allows for greater flexibility in the types of values that may be stored in each column (such as numeric, character, factor, etc.). Before performing any data analysis, it is common practice to transform a matrix to a data frame.

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

```
Pass<-function(x){
  if(x>=50){
    print("Passed")
  }
  else {
    print("Failed")
  }
}
Pass(Test$score)
```

```
## Warning in if (x >= 50) {: the condition has length > 1 and only the first
## element will be used
```

```
## [1] "Passed"
```

```
Pass<-function(x){
  ifelse(x>=50,print("Passed"),print("Failed"))
}
Pass(Test$score)
```

```
## [1] "Passed"
## [1] "Failed"
```

```
## [1] "Passed" "Passed" "Failed" "Passed"
```

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

    Answer: 'If' and 'else' did not work while 'ifelse' did. While the first option just takes one element into account and returns just the output of the first element, we have 4 elements to be tested. 'ifelse' considers each element in the list and therefore, here we shall be using this option.