Assignment 2: Coding Basics

Yanxi Krista Peng

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.
#create a sequence of numbers from one to 100 with a gap of 4, and name this sequence as HW2_seq
HW2_seq <-seq(1,100,4)

#print the sequence
HW2_seq
```

```
##  [1]  1  5  9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69 73 77 81 85 89 93 97
```

```
#2.
# print the mean and median of the sequence
mean(HW2_seq)
```

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

```
median(HW2_seq)
```

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

```
#3.
#Conditional statements
mean(HW2_seq) > median(HW2_seq)
```

```
## [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.

```
#Character
c("Krista","Van","Yennefer","Geralt") -> names.of.students
#Numeric
c(83,66,100,48) -> test.scores
#Logical
c("TRUE","TRUE","TRUE","FALSE") -> pass.or.not
```

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.

```
data.frame(Student_Name=names.of.students, Test_Scores=test.scores,Pass=pass.or.not) -> student.test
student.test
```

```
##   Student_Name Test_Scores  Pass
## 1       Krista          83  TRUE
## 2          Van          66  TRUE
## 3     Yennefer         100  TRUE
## 4       Geralt          48 FALSE
```

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

   Answer: In data frame, it can contain different types of data for both qualitative and quantitative. However, in a matrix all the elements are the same type of data.

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.

```
passing_function <- function(Scores){
  pass.or.not <- ifelse (Scores<50,F,T)
  print (pass.or.not)
}
```

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

```
passing_function(student.test$Test_Scores)
```

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

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

    Answer: 'ifelse' only apply for easy condition, for example TRUE and FALSE or 0 or 1. However, 'if' and 'else' can basically suitable for any condition.