Assignment 2: Coding Basics

Sakina Shahid

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.
seq_30 <- seq(1,30,3)
#the code above asks R to take the sequence from 1 to 30 by a space of 3

#2.

mean (seq_30)
```

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

```
median (seq_30)
```

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

```
# here I am computing the mean and median of the entire sequence using inbuilt R functions

mean <- mean(seq_30)
median <- median(seq_30)

#I am creating variable names

#3.
mean > median
```

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

```
#the statement asks whether the mean is greater than the median. The output is FALSE indicating that it
```

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.

```
names_of_students <- c("Sakina","Umi","Ivan","Abbas")
#names of students is a character vector
test_scores <- c (10,49,75,90)
#test scores are a numerical vector
pass_fail <- c(FALSE,FALSE,TRUE,TRUE)
#pass or fail are a logical vector
class(pass_fail)
```

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

```
#Step 1: Converting my vectors to data frames
names_of_students.df <- as.data.frame(names_of_students)
is.data.frame(names_of_students.df)
```

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

```
test_scores.df <- as.data.frame(test_scores)
is.data.frame(test_scores.df)
```

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

```
pass_fail.df <- as.data.frame(pass_fail)
class(pass_fail.df)
```

```
## [1] "data.frame"
```

```
school.df <- data.frame("Name"=names_of_students,"Score"=test_scores,"Pass"=pass_fail)

class(school.df)
```

```
## [1] "data.frame"
```

```
school.df
```

```
##     Name Score  Pass
## 1 Sakina    10 FALSE
## 2    Umi    49 FALSE
## 3   Ivan    75  TRUE
## 4  Abbas    90  TRUE
```

```
#I named the data frame school since all of the vectors fall under this category
```

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

   Answer: A matrix contains of only two dimensions, like a column and a row and the vectors must be of the same class (example: numerical). In comparison, a data frame can contain multiple types of classes, such as logical, numerical, character.

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.

```
school_scores <- function(x){
  grades <- ifelse (x>=50,"TRUE","FALSE")
  print(grades)
}
```

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

```
final_grades <- school_scores (test_scores)
```

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

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

   Answer: I used the `ifelse` function and it worked because of the following logic: ifelse (input condition, scenario if the condition is met, scenario if the condition is not met). In this case, x had to be greater than 50, if this was true the student would PASS, if not they would FAIL. When I applied my function to the vector, it would automatically print my results since this was inputted into the logic of my earlier function.