Assignment 2: Coding Basics

Addie Navarro

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.

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. Finding sequence (Sequence1) from 1 to 100 that increases by 4
sequence1<-seq(1,100,4)
sequence1
```

```
##  [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. Mean and Median of Sequence1, both were found to be 49
median(sequence1)
```

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

```
mean(sequence1)
```

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

```
#3. Mean of Sequence1 greater than median of Sequence1? Mean is not greater than median, conditional st
mean(sequence1)>median(sequence1)
```

```
## [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.

```
#5 Character vector of names of students
names<-c("Sansa", "Arya","Jon", "Bran")
names
```

```
## [1] "Sansa" "Arya"  "Jon"   "Bran"
```

```
#Integer vector of students' test scores
testscores<-c(95, 82, 49, 91)
testscores
```

```
## [1] 95 82 49 91
```

```
#Logical vector of whether the students passed or not
passinggrade<- testscores>50
passinggrade
```

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

```
#7 Combining vectors into a data frame
student_grades.df<-data.frame(names, testscores, passinggrade)
student_grades.df
```

```
##   names testscores passinggrade
## 1 Sansa         95         TRUE
## 2  Arya         82         TRUE
## 3   Jon         49        FALSE
## 4  Bran         91         TRUE
```

```
#8 Renaming columns of data frams
colnames(student_grades.df)<-c('Student Name', 'Test Score', 'Passed Class')
student_grades.df
```

```
##   Student Name Test Score Passed Class
## 1        Sansa         95         TRUE
## 2         Arya         82         TRUE
## 3          Jon         49        FALSE
## 4         Bran         91         TRUE
```

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

   Answer: Data frames contain different classes of data, whereas a matrix contains only one class of data.

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.

```
#10 if/else statement function
pass_fail<-function(x){ifelse(x>50, "pass", "fail")}
pass_fail(testscores)
```

```
## [1] "pass" "pass" "fail" "pass"
```

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

Answer: ‘ifelse’ worked for me better than ‘if’ and ‘else’ because it allowed me to use the logical vector of “pass” or “fail” and was more convenient and faster than using ‘if’ and ‘else’.