Assignment 2: Coding Basics

Jared Wang

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 last name into the file name (e.g., "Salk_A02_CodingBasics.Rmd") prior to submission.

The completed exercise is due on Tuesday, January 21 at 1:00 pm.

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.

```r
#1. a sequence from 1 to 100 by 4
seq_1.100.4 <- seq(1, 100, 4) # from, to, by

#2. mean and median of the sequence
mean(seq_1.100.4)
```

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

```r
median(seq_1.100.4)
```

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

```r
#3. is the mean larger than the median?
mean(seq_1.100.4) > median(seq_1.100.4)
```

```
## [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.

```r
#create vectors
vec_name <- c("bill","eliza","katherine","ted") #character
vec_score <- c(34,70,83,66) #numeric
vec_pass <- c(vec_score > 50) #logical

#merge vectors into a data frame
df_exam <- data.frame(vec_name, vec_score, vec_pass) #
names(df_exam) <- c("name","score","passed")
```

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

   Answer: a data frame can include different types of data, but a matrix can only contain one type 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.

```r
#-----------------------ifelse-----------------------
#write the ifelse function
func_pass <- function(x){
  output <- ifelse(x >= 50, TRUE, FALSE)
  print(output)
}

#run the function
output_pass <- func_pass(vec_score)
```

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

```r
#-----------------------if/else-----------------------
#write the if/else function
func_pass_b <- function(x) {
  output_b <- if(x >= 50) {
    TRUE
  }
  else if (x < 50) {
    FALSE
  }
  print(output_b)
}

#run the function
output_pass_b <- func_pass_b(vec_score)
```

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

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

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

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

Answer: only the ifelse function worked, because the `if` and `else` functions only process the first element of a vector.