Assignment 2: Coding Basics

Jack Eynon

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.

```
#1. Creating sequence from 1 to 100 by 4.
answer1 <- seq(1,100,4)
#2. Taking mean and median of sequence. Mean = 49. Median = 49
mean(answer1)
```

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

```
median(answer1)
```

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

```
summary(answer1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1      25      49      49      73      97
```

```
#3. Logical test --> FALSE; mean is not greater than median.
mean(answer1) > median(answer1)
```

```
## [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.

```
names <- c("Alec", "Patrick", "David", "Jimmy") ## Character vector
scores <- c(95, 49, 87, 73) ## Numeric vector
passed <- c(TRUE, FALSE, TRUE, TRUE) ## Logical vector
test.results <- data.frame(names, scores, passed) ## Data frame combining the 3 vectors
names(test.results) <- list("Names", "Scores", "Passed") ## assigning names to column headers
```

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

   Answer: A matrix is a 2-dimensional structure that contains elements of the same type. A data frame is also a 2-dimensional structure, but it can contain different element types (like character, numeric, and logical elements).

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.

```
passing.grade <- function(x) {
  ifelse(x>=50, print(TRUE), print(FALSE))    ## Creating function with ifelse command
}
passing.grade(test.results$Scores) ## This works
```

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

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

```
alternative <- function(x) {   ## Creating function using 'if' and 'else' commands
  if(x < 50) {
    print(FALSE)
  }
  else {
    print(TRUE)
  }
}
alternative(test.results$Scores) ## Only performs logical test on first element of vector
```

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

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

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

    Answer: Only the ifelse option worked. For the 'if' and 'else' option, the function was only applied to the first element in the vector (i.e. the first test score).