

Assignment 2: Coding Basics

Devin Domeyer

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. I am telling R to create a sequence from 1 to 100, increasing by 4.  
sequence <- seq(1, 100, 4)  
sequence
```

```
## [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. I am telling R to find the mean and median of the sequence  
mean <- mean(sequence)  
median <- median(sequence)
```

```
#3. The output of this conditional statement will either be true or false.  
mean > median
```

```
## [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.

```
# The vector "name" is numeric
name <- c("Todd", "Sally", "Ron", "Sandra")
# The vector "score" is character
score <- c(42, 85, 63, 94)
# The vector "pass_fail" is logical
pass_fail <- c(FALSE, TRUE, TRUE, TRUE)

# Now I combine the new vectors into a single dataframe using cbind
test_stats <- cbind(name, score, pass_fail)

# Now I change the column names
colnames(test_stats) <- c("Student_Name", "Test_Score", "Pass/Fail")

test_stats
```

```
##      Student_Name Test_Score Pass/Fail
## [1,] "Todd"      "42"       "FALSE"
## [2,] "Sally"     "85"       "TRUE"
## [3,] "Ron"       "63"       "TRUE"
## [4,] "Sandra"    "94"       "TRUE"
```

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

Answer:

A data frame is a more general form of data structure than a data matrix. In a matrix all the columns have the same mode and the same length. In a data frame, you can have different modes.

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.

```
pass_test <- function(x) {
  ifelse(x>50, TRUE, FALSE)
}

pass_test(score)
```

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

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

Answer:

`ifelse` worked because I put in my logical expression first “if x is greater than 50”, then if that expression matched the vector value it was “TRUE” and if it didn’t it was “FALSE”. It would have worked if I separated out the “if” and “else” function elements but I wanted to keep my code clean.