

Assignment 2: Coding Basics

Robert Hill

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 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.  
seq <- seq(1,100,4)  
seq(1,100,4) #built a sequence of numbers from 1-100, increasing by 4's & assigned the sequence to 'seq'
```

```
## [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 <- mean(seq)  
median <- median(seq)  
mean #mean of sequence
```

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

```
median #median of sequence
```

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

```
#3.  
mean>median #Is the mean of the sequence greater than the 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.

```
Student_Name <- c("Nemo", "Dory", "Marlin", "Bruce") #vector type: character  
Test_Score <- c(90, 70, 50, 30) #vector type: numeric  
Passed <- Test_Score>=50 #vector type: logical  
Exam_Results <- data.frame("Student Name"=Student_Name,  
  "Test Score (out of 100)"=Test_Score,"Passed"=Passed)
```

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

Answer: Data frames can contain different types of data; like characters, numbers, etc; while matrices cannot.

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

```
PassFail <- function(x){  
  ifelse(x>=50,"PASS","FAIL")  
}  
grades <-PassFail(Test_Score)  
print(grades)
```

```
## [1] "PASS" "PASS" "PASS" "FAIL"
```

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

Answer: 'ifelse' worked while 'if' and 'else' did not work. When running the latter function, an error of "the condition has length > 1". So an if() function can only evaluate 1 element, while an ifelse() can evaluate multiple elements.