

Assignment 2: Coding Basics

Sara Diamond

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.

```
seq1 <- seq(1, 100, 4) # assigning sequence a name
seq1 # seeing the 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.

```
mean(seq1) #finding the mean of the sequence
```

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

```
median(seq1) #finding the median of the sequence
```

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

#3.

```
mean(seq1) > median(seq1) #findinf the median and mean of the sequence
```

```
## [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 and 6
#creating vectors
vector1 <- c("Sara", "Bob", "Susan", "Frank") #character vector
vector1

## [1] "Sara" "Bob" "Susan" "Frank"

vector2 <- c(98, 75, 45, 84) #numeric vector
vector2

## [1] 98 75 45 84

vector3 <- c(TRUE, TRUE, FALSE, TRUE) #logical vector
vector3

## [1] TRUE TRUE FALSE TRUE

#7
dataframe1 <- data.frame(vector1,vector2,vector3) #assigning to a data frame

#8
names(dataframe1) <- c("Student Name","Grade","Passed"); print(dataframe1)

##   Student Name Grade Passed
## 1      Sara     98    TRUE
## 2       Bob     75    TRUE
## 3     Susan     45   FALSE
## 4      Frank     84    TRUE

#renaming the columns inside the data frame
```

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

Answer:

A data frame can have columns of different vector types such as numeric, logical, and character. A matrix can only have vectors of the same type. Both however, are 2 dimensional.

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
#creating the if and else statement for test scores
test.score <- function(x) {
  if(x >= 50) {
```

```

    TRUE
  }
  else {
    FALSE
  }
}

test.score(vector2) #error when running if and else

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

## [1] TRUE
#creating the ifelse statement for test scores vector
test.score2 <- function(x){
  ifelse(x >= 50, "TRUE", "FALSE") #log_exp, if TRUE, if FALSE
}

print(test.score2(vector2)) #making the values visible

## [1] "TRUE" "TRUE" "FALSE" "TRUE"

```

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

Answer:

The ifelse function worked and the if and else one did not. This is because ifelse checks the entire vector whereas using the if and else only checks the first part of the vector.