

# Assignment 2: Coding Basics

Grace Randall

## 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, Part 1

1. Generate a sequence of numbers from one to 30, increasing by threes. 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.
question_one <- seq(1,30,3)#forming a sequence from 1 to 30 with jumps of 3
question_one #checking output of this sequence

## [1] 1 4 7 10 13 16 19 22 25 28

#2.
mean_q_one <- mean(question_one) # calculating the mean of question_one
median_q_one <- median(question_one) # calculating the median of question_one
mean_q_one #printing the mean

## [1] 14.5
median_q_one #printing the median

## [1] 14.5

#3.
(mean_q_one>median_q_one)# checking if mean is greater than median

## [1] FALSE
```

## Basics, Part 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.

```
name <- c("Alice", "Bob", "Charlie", "Dan", "Eve") #character string vector of students
name <- as.data.frame(name)
score_v <- c(35,73,73,17,31) # number vector or scores
score <- as.data.frame(score_v)
did_they_pass <- c(FALSE,TRUE,TRUE,FALSE,FALSE) # Boolean vector
did_they_pass <- as.data.frame(did_they_pass)
test_results <- cbind(name,score,did_they_pass)
class(test_results)
```

```
## [1] "data.frame"
```

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

Answer: If this were held in a matrix, then all of the column's would need to be of the same data type. in this case they would all be turned into chr.

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.

```
pass_or_fail <- function(scores){ #create function

for(x in 1:length(scores)){ # loops for all of the values given in the vector
ifelse (scores[x]>=50, print(TRUE), print(FALSE)) #checks if passes or fails
}}
pass_or_fail(score_v) #testing function on answer from 5
```

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

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

Answer: Both ifelse or if and else would work for this. Since there are only two options for what to do that can be defined by a single conditional ifelse is sufficient for this function. Anything that can be written using ifelse can also be written using if and else.