

# Assignment 2: Coding Basics

Karly Nocera

## 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.
### sequence (from, to, by)

seq_by4 = seq(1, 100, 4)
seq_by4

## [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.
### use the mean and median functions on the named sequence

mean(seq_by4)

## [1] 49

median(seq_by4)

## [1] 49

#3.
### > asks if mean is greater than median

mean(seq_by4) > median(seq_by4)

## [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. & 6.
### use command <- c() to create vector

students <- c("jack", "diane", "bruce", "clarence")
# character vector
# use "" for string/names

score <- c(46, 89, 72, 98)
# numeric vector

test_pf <- c(FALSE, TRUE, TRUE, TRUE)
# logical vector
# manually enter based on score

#7. & 8.

school_df <- data.frame("Student" = students, "Score" = score, "Passed" = test_pf)
### use command data.frame to assign "column names" to vector

school_df

##      Student Score Passed
## 1      jack    46  FALSE
## 2      diane    89   TRUE
## 3      bruce    72   TRUE
## 4 clarence    98   TRUE

### final data frame
```

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

Answer: Whereas a matrix is made of all the same element, this data frame is not uniform in type, it includes string/character, numeric, and logic entries.

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.

```
#11.

### create function x called "pass_grade" such that if x is greater than
# 50 print as True, otherwise print as false

pass_grade <- function(x){
```

```
    ifelse(x>50, "True", "False")
}

student_test_result <- pass_grade(score) #score is the vector of student scores to evaluate
student_test_result
```

```
## [1] "False" "True"  "True"  "True"
```

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

Answer: `ifelse` worked because it responds to a logical vector whereas `if` and `else` only react to one logical element (runs the function for only the first score rather than each).