

# Assignment 2: Coding Basics

*Nikki Egna*

## 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 last name into the file name (e.g., “Salk\_A02\_CodingBasics.Rmd”) prior to submission.

The completed exercise is due on Tuesday, January 21 at 1:00 pm.

## 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_factor4<-seq(1,100,4) #Sequence (from, to, by)
```

```
#2.  
mean(seq_factor4) #Use mean function to calculate mean of sequence
```

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

```
median(seq_factor4) #Use median function to calculate median of sequence
```

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

```
#3.  
mean(seq_factor4)>median(seq_factor4) #Ask R if the mean of the sequence is 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.

```
a<-c("Ashley","Brittany","Cameron","Danny") #Creating a character vector of names
b<-c(70,99,40,85) #Create a numeric vector with the test scores
c<-c(TRUE,TRUE,FALSE,TRUE) #Create a logical vector of whether or not the test score was a passing grade

student_grades<-data.frame(a,b,c)
colnames(student_grades)<-c("Name","Grade","Passing")
```

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

Answer: It contains numeric, character, and logical data whereas matrices can contain only one type of data

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.

```
passing<-function(x){
  y<-ifelse(x$Grade>50,TRUE,FALSE)
  print(y)
}
passing(student_grades)
```

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

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

Answer: `ifelse` worked because there are two options that the answer could be, either TRUE or FALSE. So you need to give it a way to assign the answer if the condition is met with the `if`, and if the condition is not met with the `else`. So only `ifelse` works because you need the two different options for the two different outcomes.