# Assignment 2: Coding Basics

Aditi Jackson

## 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.
seq(1,30,3) # generating sequence of numbers from 1 to 30, increasing by 3's
```

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

```
one_to_thirty_by_three <- seq(1,30,3) # assigning name to sequence
one_to_thirty_by_three # running variable that contains sequence created above
```

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

```
#2.
mean(one_to_thirty_by_three) # calculating mean of sequence from step 1
```

```
## [1] 14.5
```

```r
median(one_to_thirty_by_three) # calculating median of sequence from step 1
```

```
## [1] 14.5
```

```r
#3.
mean(one_to_thirty_by_three) > median(one_to_thirty_by_three) # testing if mean > 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.

```r
#5 & $6.
Name_vector <- c("Anna", "Ben", "Cameron", "Denis") # Vector with student names (string)
Test_score_vector <- c(99,49,97,48) # Vector with test scores (integer)
Passed_failed_vector <- c(TRUE,FALSE,TRUE,FALSE) # Vector noting passage or failure (Boolean)

#7.
df_student_scores <- as.data.frame(cbind(Name_vector,Test_score_vector,Passed_failed_vector)) #combinin
df_student_scores # running this line with show the data frame in the console
```

```
##   Name_vector Test_score_vector Passed_failed_vector
## 1        Anna                99                 TRUE
## 2         Ben                49                FALSE
## 3     Cameron                97                 TRUE
## 4       Denis                48                FALSE
```

```r
#8.
df_student_scores_COPY <- df_student_scores # making copy of original data set in order to apply new co
df_student_scores_COPY # running this line will generate a copy of the data frame in the console
```

```
##   Name_vector Test_score_vector Passed_failed_vector
## 1        Anna                99                 TRUE
## 2         Ben                49                FALSE
## 3     Cameron                97                 TRUE
## 4       Denis                48                FALSE
```

```r
names(df_student_scores_COPY) <- c("Student Name","Test Score","Passed Test? ") # using names() to chan
df_student_scores_COPY # running this line with generate the data frame with new headers applied
```

```
##   Student Name Test Score Passed Test?
## 1         Anna         99          TRUE
## 2          Ben         49         FALSE
## 3      Cameron         97          TRUE
## 4        Denis         48         FALSE
```

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

   Answer: This data frame contains vectors with different modes, which makes it different from a matrix. A data frame is more general than a matrix. While a data frame can comprise vectors with different modes (numbers and characters), all vectors in a matrix must contain the same mode (only numbers or only characters).

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.

```r
#10.
x <- Test_score_vector # assigning test score vector to "x" for easier manipulation

# creating function using an ifelse statement
passing_grade_function <- function(x){
  report_grade <- ifelse(x>=50,TRUE,FALSE)
   print(report_grade)
}

# running newly created function, which prints whether or not the grades from the test score vector qua
passing_grade_function(x)
```

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

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

    Answer: The option 'ifelse' worked while using 'if' and 'else' separately did not. This is because 'ifelse' is a "vectorized" form of an "if-else" statement, meaning 'ifelse' can accept a vector while 'if-else' can only accept a variable at a time.