

# Assignment 2: Coding Basics

Ayoung Kim

## 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. Sequence 1 (I defined seq(1,30,3) as sequence 1)  
seq(1,30,3)
```

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

```
sequence1<-seq(1,30,3)
```

```
#2. Mean and Median (I got mean and median of Sequence 1 that I defined above)  
mean(sequence1)
```

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

```
median(sequence1)
```

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

```
#3. Mean > Median ? (I got 14.5 for both mean and median)  
14.5>14.5
```

```
## [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.

```
#test score  
  
#Name of Student  
student = c("Anne","Marie","Jack","Wilson")  
  
#Test score  
testscore = c(45,70,80,90)  
  
#Pass or Fail  
pass<-(testscore>50)  
  
#Test score/student name/pass  
test_score<-testscore  
student_name <-student  
pass<-pass  
  
#Data frame for Test Score  
class(test_score)
```

```
## [1] "numeric"
```

```
df_test_score <- as.data.frame(test_score)  
df_test_score
```

```
##   test_score  
## 1         45  
## 2         70  
## 3         80  
## 4         90
```

```
#Data frame for Student Names  
class(student_name)
```

```
## [1] "character"
```

```
df_student_name <-as.data.frame(student_name)
df_student_name
```

```
##      student_name
## 1           Anne
## 2           Marie
## 3           Jack
## 4           Wilson
```

```
#Data frame for 'Pass or Fail'
class(pass)
```

```
## [1] "logical"
```

```
df_pass <-as.data.frame(pass)
df_pass
```

```
##      pass
## 1 FALSE
## 2  TRUE
## 3  TRUE
## 4  TRUE
```

```
#Adding Columns
df<-cbind(df_student_name,df_test_score,df_pass)
class(df)
```

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

```
df
```

```
##      student_name test_score  pass
## 1           Anne         45 FALSE
## 2           Marie         70  TRUE
## 3           Jack         80  TRUE
## 4           Wilson         90  TRUE
```

```
names(df)<-"Test Scores of Students"
```

```
#Name
names(df_student_name)<-"Student"
names(df_test_score)<-"Score"
names(df_pass)<-"Pass"
df
```

```
##      Test Scores of Students NA      NA
## 1                Anne 45 FALSE
## 2                Marie 70  TRUE
## 3                Jack 80  TRUE
## 4                Wilson 90  TRUE
```

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

Answer: Answer: with a data frame, I can combine data sets and show in a one chart.

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.

#Function for Marie, Jack, and Wilson df\_test\_score<-x if(x>50){ "Pass" } #Function for Anne  
df\_test\_score<-x if(x<50){ 'fail' } 11. Apply your function to the vector with test scores that you created in number 5.

```
#Test score for Anne
df_test_score <-45
if (df_test_score<50){
  'Fail'
}
```

```
## [1] "Fail"
```

```
#Test score for Marie
df_test_score<-70
if(df_test_score>50){
  'pass'
}
```

```
## [1] "pass"
```

```
#Test score for Jack
df_test_score <-80
if(df_test_score>50){
  'pass'
}
```

```
## [1] "pass"
```

```
#Test score for Wilson
df_test_score<-90
if(df_test_score>50){
  'pass'
}
```

```
## [1] "pass"
```

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

Answer:The 'if' option worked for me. I used different function for Anne's score.I put x<50 for Anne's score and x>50 for the scores of Marie, Jack, and Wilson.I thought, for Anne's score, the combination of 'if' and 'else' would work, but it didn't. so I decided to put x<50 for Anne's case.