

Assignment 2: Coding Basics

John Rooney

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 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.

```
#Basics Day 1 A02 Questions
```

```
#1. #generate sequence from one to one hundred by fours
```

```
assignment_sequence<-seq(1,100,4)
assignment_sequence
```

```
## [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. #calculate mean and median of sequence
mean(assignment_sequence)
```

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

```
median(assignment_sequence)
```

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

```
#3. #test whether mean is greater than median
mean(assignment_sequence) > median(assignment_sequence)
```

```
## [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.

```
#Basics Day 2 A02 Questions
student_names <- c("John", "Jacob", "Jingleheimer", "Schmidt") #character vector
test_scores <- c(100, 75, 50, 45) #dbl vector
pass_test <- c(TRUE, TRUE, TRUE, FALSE) #logical vector
```

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.

```
#combine vectors into a data frame
test_scores_df<-data.frame(student_names, test_scores, pass_test)
test_scores_df
```

```
##   student_names test_scores pass_test
## 1      John      100      TRUE
## 2     Jacob       75      TRUE
## 3 Jingleheimer       50      TRUE
## 4     Schmidt       45     FALSE
```

8. Label the columns of your data frame with informative titles.

```
#label columns in data frame
colnames(test_scores_df)<-c("Student Name", "Test Score", "Passing")
test_scores_df
```

```
##   Student Name Test Score Passing
## 1      John      100      TRUE
## 2     Jacob       75      TRUE
## 3 Jingleheimer       50      TRUE
## 4     Schmidt       45     FALSE
```

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

Answer: This data frame is different from a matrix because it contains multiple vector types. A matrix is comprised of only a single vector type.

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.

```
#create new function
did_student_pass<-function(test_scores){
  ifelse(test_scores>=50, TRUE, FALSE)
}
```

11. Apply your function to the vector with test scores that you created in number 5.

```
#run new function
did_student_pass(test_scores)
```

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

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

Answer: ‘ifelse’ worked for me after I failed to get ‘if’ and ‘else’ to work. Honestly, I’m not sure why. My understanding of ‘ifelse’ is that it’s essentially a simpler way of coding ‘if’ and ‘else’ code, so my guess is I just goofed on my initial coding.