

Assignment 2: Coding Basics

Xiaoge Zhang

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.

```
#1. generate a sequence of numbers from 1 to 100 by 4
seq1 = seq(from=1, to=100, by=4)
#2. calculate the mean and median
mean(seq1)
```

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

```
median(seq1)
```

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

```
#3. check if the mean is greater than the median
mean(seq1) > median(seq1)
```

```
## [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.
# string
names <- c('rory','christy', 'xiaoge', 'ari')
# integer
scores <- c(90, 95, 49, 91)
# boolean
pass <- c(TRUE, TRUE, FALSE, TRUE)

#7.
score_data <- as.data.frame(cbind(names, scores, pass))

#8.
colnames(score_data) <- c('student_name', 'test_core', 'pass')
```

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

Answer: data frames can contain different types of data, while matrices can only contain a single type of data.

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.

```
#10.
check_score <- function(score_vec){
  ifelse(score_vec >= 50, TRUE, FALSE)
}

#11.
check_score(scores)
```

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

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

Answer: ifelse worked better since it takes a vectorized input and gives a vectorized output. If I had used 'if' and 'else', I'd have to do a for loop to go through each item in the vector input.