

Assignment 2: Coding Basics

Yunting Wang

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 first and last name into the file name (e.g., “FirstLast_A02_CodingBasics.Rmd”) prior to submission.

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.
sequence1<-seq(1,100,4) #create a sequence from 1 to 100, increase by 4
sequence1

## [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.
mean(sequence1) #calculate mean

## [1] 49

median(sequence1) #calculate median

## [1] 49

#3.
mean(sequence1)>median(sequence1) #whether mean is greater than 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.

```
student_name <- c("A","B","C","D","E") #student name
score <- c(38,77,92,49,84) #test score
pass_or_fail <-c(score>50) #whether pass test
grading_chart= data.frame("name"=student_name,"score"=score,"passing"=pass_or_fail)
grading_chart
```

```
##   name score passing
## 1    A    38   FALSE
## 2    B    77    TRUE
## 3    C    92    TRUE
## 4    D    49   FALSE
## 5    E    84    TRUE
```

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

Answer: Matrix can only contain 1 data type, whereas data frame can store different data types in different column.

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_test1=function(x){
  ifelse(x>=50,"True","False")
}
passing_test2=function(x){
  if (x>=50){
    print("True")
  }
  else {
    print("False")
  }
}
passing_test1(score)
```

```
## [1] "False" "True"  "True"  "False" "True"
```

```
passing_test2(score)
```

```
## Warning in if (x >= 50) {: the condition has length > 1 and only the first
## element will be used
```

```
## [1] "False"
```

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

Answer: `ifelse` worked. `if` and `else` doesn't accept vector, so it can only print the first element.