

Assignment 2: Coding Basics

Joshua Frear

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.

```
#Question 1. seq() can take 3 arguments, including the increment by which to increase.
```

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

```
## [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
```

```
#Question 2. calculating the mean and median of the sequence from 1
```

```
mean(seq_one)
```

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

```
median(seq_one)
```

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

```
#Question 3. ">" runs a True/False test on the expression.
```

```
mean(seq_one) > median(seq_one)
```

```
## [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.

```
#text vector for student names
student_names <- c("Abe", "Beth", "Donald", "Toni")
#integer vector for test scores
student_scores <- c(95, 82, 49, 98)
#logical vector for whether a student passed
student_passing <- c(TRUE, TRUE, FALSE, TRUE)

student_records_df <- data.frame("Name"=student_names,
  "Test_Score"=student_scores, "Passing"=student_passing)
```

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

Answer: It has non-numeric columns, including one with text values and one with logical values.

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.

```
#trying a function with if() and else
check_pass <- function(x){
  if(x >= 50) {
    print(TRUE)
  }
  else{
    print(FALSE)
  }
}

#trying a function with ifelse()
check_pass_ifelse <-function(x){
  ifelse(x >= 50, TRUE, FALSE)
}

#running the ifelse function on student_scores from before
check_pass_ifelse(student_scores)
```

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

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

Answer: 'ifelse' worked with all four students. 'if' did not work, because apparently it is not set up to handle vectors by default, so it only returned the answer for the first item in the vector. I did not expect this!