Assignment 2: Coding Basics

Catherine Otero

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.Generate a sequence of numbers from one to 100, increasing by fours. Name the output quad_interval
quad_interval <- seq(1, 100, 4)
quad_interval
```

```
##  [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 the mean and median of quad_interval
mean(quad_interval)
```

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

```
median(quad_interval)
```

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

```
#3. Is mean of quad_interval greater than median of quad_interval? Answer=FALSE
mean(quad_interval) > median(quad_interval)
```

```
## [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. Helpful link https://www.geeksforgeeks.org/creating-a-data-frame-from-vectors-in-r-programming/

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

```
#5. Vector creation. Helpful link https://techvidvan.com/tutorials/r-vector/

#character vector
student_names <- c("Maria", "Cat", "Carlos", "Quincy")

#numeric vector
test_scores <- c(100, 95, 93, 90)

#logical vector
passing_score <- c(TRUE, TRUE, TRUE, TRUE)

#7. Create data frame from vectors
exam_results.df<- data.frame(student_names, test_scores,
                      passing_score); exam_results.df
```

```
##   student_names test_scores passing_score
## 1         Maria         100          TRUE
## 2           Cat          95          TRUE
## 3        Carlos          93          TRUE
## 4        Quincy          90          TRUE
```

```
#8. renaming columns in data frame. Helpful link https://www.geeksforgeeks.org/change-column-name-of-a-
exam_results.df <-data.frame(student_names, test_scores,
                      passing_score)

colnames(exam_results.df) <- c('Student Names','Test Scores','Passing Score?'); exam_results.df
```

```
##   Student Names Test Scores Passing Score?
## 1         Maria         100           TRUE
## 2           Cat          95           TRUE
## 3        Carlos          93           TRUE
## 4        Quincy          90           TRUE
```

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

   Answer: This dataframe contains different types of data (i.e. numeric and character data) while a matrix can only contain one kind of data, usaually numeric.

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.

```r
#checking if students passed with if and else function
check_passing <- function(x) {
  if(x >= 50) {
    x = TRUE
  }
  else if (x < 50) {
    x = FALSE
  }
}

check_students_passed <- check_passing('passing_score'); check_students_passed
```

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

```r
#checking if students passed with ifelse function

check_passing2 <- function(x){
  ifelse(passing_score, x>=50, x<50)
}

check_students_passed2 <- check_passing2('passing_score'); check_students_passed2
```

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

Helpful links: https://www.datacamp.com/community/tutorials/if-else-function-r

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

    Answer: ifelse works because it checks the whole vector and returns outputs for each row of data in the vector while if and else only outputs one answer.