Assignment 2: Coding Basics

Amanda Braun

## 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 last name into the file name (e.g., “Salk\_A02\_CodingBasics.Rmd”) prior to submission.

The completed exercise is due on Tuesday, January 21 at 1:00 pm.

## 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. seq is function to generate a sequence, (from, to, by) why doens't it produce the numbers?   
amanda\_sequence <- seq(1, 100, 4)   
  
#2.   
mean(amanda\_sequence)

## [1] 49

median(amanda\_sequence)

## [1] 49

#3.   
mean(amanda\_sequence) < median(amanda\_sequence)

## [1] FALSE

## Basics Day 2

1. 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.
2. Label each vector with a comment on what type of vector it is.
3. Combine each of the vectors into a data frame. Assign the data frame an informative name.
4. Label the columns of your data frame with informative titles.

names <- c("Amanda", "Rani", "Cate", "Rachel")   
names

## [1] "Amanda" "Rani" "Cate" "Rachel"

#character vector   
testscore <- c(95, 48, 92, 72)  
testscore

## [1] 95 48 92 72

#numeric vector   
testpass <- c(TRUE, FALSE, TRUE, TRUE)  
testpass

## [1] TRUE FALSE TRUE TRUE

#logical vector   
  
classscoresNovember10 <- data.frame(names, testscore, testpass)  
classscoresNovember10

## names testscore testpass  
## 1 Amanda 95 TRUE  
## 2 Rani 48 FALSE  
## 3 Cate 92 TRUE  
## 4 Rachel 72 TRUE

names(classscoresNovember10) <- c("Name", "Score", "Pass"); View(classscoresNovember10)

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

Answer: Both data frames and matrixes represent two dimensional data sets. A data frame can display multiple types of data - character, logical, numeric. A matrix represents one type of data in a set number of rows and columns.

1. 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.
2. Apply your function to the vector with test scores that you created in number 5.

passingscore <- function(x){  
 if (x < 50) (x=FALSE) else (x=TRUE)} # if and else function without print   
passingscore (45)

## [1] FALSE

passingscore(testscore)

## Warning in if (x < 50) (x = FALSE) else (x = TRUE): the condition has length > 1  
## and only the first element will be used

## [1] TRUE

passingscoreprint<- function (x) {if (x<50) print("True") else print("False")} # "if" and "else" function with print   
passingscoreprint(45) #testing single object

## [1] "True"

passingscoreprint(testscore) #testing testscore vector

## Warning in if (x < 50) print("True") else print("False"): the condition has  
## length > 1 and only the first element will be used

## [1] "False"

passing\_score2print <- function(x) {  
 ifelse((x<50),   
 print("False"), print("True"))} #ifelse function with print   
passing\_score2print(6)

## [1] "False"

## [1] "False"

passing\_score2print(testscore)

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

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

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

Answer: #“if” and “else” work for one value, but do not work for a vector. When “if” and “else” is used for a vector, an warning message is produced that the conditions has a length > 1 and only the first element will be used, meaning that we cannot input a vector with multiple objects into the function. #“ifelse” does work for a vector both a single object and for a vector