

SimonHeinberg_A02_CodingBasics.Rmd

Simon_Heinberg

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, Part 1

1. Generate a sequence of numbers from one to 30, increasing by threes. Assign this sequence a name.

```
sequence30 <- (seq(0, 30, 3)) #I created an object named 'sequence30' that displays a sequence of numbers
```

2. Compute the mean and median of this sequence.

```
meanfunction <- mean(sequence30)
medianfunction <- median(sequence30) #I created an object named 'meanfunction' that calculates the mean
```

3. Ask R to determine whether the mean is greater than the median.

```
meanfunction > medianfunction #I asked R to give a 'TRUE' or 'FALSE' answer to whether the output of 'meanfunction' is greater than the output of 'medianfunction'

## [1] FALSE
```

4. Insert comments in your code to describe what you are doing.

I created an object named ‘sequence30’ that displays a sequence of numbers from 0 to 30 by intervals of 3

I created an object named ‘meanfunction’ that calculates the mean of ‘sequence30’ and an object named ‘medianfunction’ that calculates the median of ‘sequence30’

I asked R to give a ‘TRUE’ or ‘FALSE’ answer to whether the output of ‘meanfunction’ is greater than the ‘medianfunction’

Basics, Part 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.

```
students <- c("jill", "jack", "james", "joy")
scores <- c(35, 90, 72, 95)
pass_fail <- scores > 50
```

6. Label each vector with a comment on what type of vector it is.

```
students #vector of student names
```

```
## [1] "jill" "jack" "james" "joy"
```

```
scores #vector of student scores
```

```
## [1] 35 90 72 95
```

```
pass_fail #vector of whether or not they have passed the test
```

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

7. Combine each of the vectors into a data frame. Assign the data frame an informative name.

```
df <- data.frame(students, scores, pass_fail)
```

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

```
names(df) <- c("studentnames", "studentscores", "did_the_student_pass")
```

After failing to find a function for renaming the columns of a dataframe, I asked chatGPT which function I should use to rename a dataframe. I found the `names(df)` function through chatGPT.

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

Answer: The data frame is different from a matrix because it has three types of data, whereas a matrix only has one 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.

```
pass_fail_function <-function(x) {ifelse(x>50, print("pass"), print ("fail"))}
```

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

```
pass_fail_output <- pass_fail_function(scores)
```

```
## [1] "pass"  
## [1] "fail"
```

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

Answer: 'ifelse' worked since it ran the function for all four objects in the vector, calculating the expression `x>50` for each object in the vector. 'if' and 'else' should also work since it is the longer notational form of the 'ifelse' function.