

Heather Siart  
ECO-634 – Environmental Data Analysis Lab  
9/15/2021  
Lab Help: Bonnie, John, Matt, Mandy

## **R Fundamentals 1**

Q1: The two outputs are different because `c(1, 2, 3)` is numeric and `"c(1, 2, 3)"` is a character string.

Q2: `c_1` is a variable because you are storing a value in `c_1` that can later be easily accessed by using the variable's name.

Q3: `c_2` is a variable because again we are storing a value that can easily be recalled in R.

Q4: `c_1` and `c_2` have different values because `c(1, 2, 3)` has a numeric value and was assigned to `c_1`, and `"c(1, 2, 3)"` has a character value and was assigned to `c_2`.

Q5: The matrix is 3x1. There are three rows and one column.

Q6: `mat_1[3, 1]` returns the value 3

Q7: `mat_2 <- matrix(my_vec, nrow = 2, ncol = 3)`

Q8: `mat_3 <- matrix(my_vec, nrow = 3, ncol = 2)`

Q9: R used rows to fill in the matrix.

Q10: `mat_4 <- matrix(my_vec, nrow = 4, ncol = 4)`

Q11: R gives the warning "... data length [3] is not a sub-multiple or multiple of the number of rows [4]". R repeats "1, 2, 3" across the rows but because it is not a multiple of three when it ends and goes to start a new row it picks up from where it left off.

Example: 1, 2, 3, 1  
          2, 3, 1, 2

Q12:

`my_list_1[[1]]`

A. value (5.2)

B. Subsetting Operation: `[[ ]]`

C. Calling `[[1]]` is asking R for the position of 1.

`my_list_1[[as.numeric("1")]]`

A. value (5.2)

B. Subsetting Operation: `[[ ]]`

C. `[[as.numeric("1")]]` is calling for the element under "1" but instead of returning NULL it gives the value of 5.2 because we have asked for the character "1" to be read as a numeric value.

`my_list_1[["1"]]`

A. NULL

B. Subsetting Operation: `[[ ]]`

C. The square brackets are calling for the element in the first row but we are giving it a character value and R can not change it to a numeric without being told to do so.

`my_list_1[["one"]]`

A. value (5.2)

B. Subsetting Operation: `[[ ]]`

C. Using `[[ ]]` for the subsetting operator it is calling for the element in row one by name.

`my_list_1$one`

A. value (5.2)

B. Subsetting Operation: `$`

C. Because you are calling `$` as the operator, it is calling the value from row one by name.

`my_list_1$"one"`

A. value (5.2)

B. Subsetting Operation: `$`

C. `$` is calling "one" which is the row name for that value

`my_list_1$1`

A. error: unexpected numeric constant in "my\_list\_1\$1"

B. N/A

C. N/A

`my_list_1$"1"`

A. NULL

B. Subsetting Operation: `$`

C. When you use `$` to call for the row name "1" but you get NULL as a response because we are not asking R to read it as numeric and it is reading it as a character.

Q13: None of these lines gave "five point two" as a result because none of the lines are asking for that information, they are only asking from row one. If we wanted "five point two" as a result we would have to ask for row two like, `my_list_1[[2]]`.

Q14: `my_list_1[["1"]]` and `my_list_1$"1"` returned NULL because there is nothing in the list that is a character value of "1" and R can not change a numeric value to a character value.