

## Lab 2 R Fundamentals 2

Jackie Stephens

**Q1 (2 pts.):** Show the R code you used to create `vec_2`.

```
n = 12345
vec_1 = sample(12, n, replace = TRUE)
head(vec_1)
vec_1
vec_2 = vec_1 == 3
vec_1[vec_2]
```

**Q2 (2 pts.):** Give two reasons why determining which elements in `vec_1` have value 3 by visual inspection is a bad idea.

1. There are 12345 values to read through so it will be time consuming.
2. Again, since there are so many values, you could lose your place or count.

**Q3 (1 pt.):** Why didn't you always get the same count of 3 entries each time?

The count of 3 entries was not the same each time because the vector provides 10 outputs that each have 1/12 the chance of being the value 3.

**Q4 (3 pts.):** Considering the different vectors generated each time, explain why using a logical test is a safe way to select entries with a value of 3.

A logical test is a safe way to select entries with a value 3 because with each generated number of entries you can start developing expectations and can bias your own counts if you do hand counts. Logical tests guarantee accuracy with each vector generated.

**Q5 (5 pts.):** Explain why performing logical 'by hand' subsetting is very very bad practice. You may want consider re-usability of code, working with different sized data sets, and sharing code with collaborators.

- Your answer should cite at least *two* reasons why 'by hand' subsetting is bad.

Performing a logical "by hand" sub setting" is a very bad practice because of human error can lead to inaccurate results.

1. An individual trying to use your code for reference will not understand where you got your information from if it is not in the code.
2. The larger the data set you use the harder it will be to accurately count or complete the subsetting task "by hand". You might not be able to see all of the values in one window.
3. Your word is not "enough" evidence that it is correct.

**Q6 (3 pts.):** Provide the code for your modified loop. It must run as a self-contained example on a fresh R session on my computer.

```
for (i in 1:10) {
  print(
    paste0(
      "This is loop iteration: 1"))
}
```

**Q7 (2 pts.):** Provide the code for the modified loop that executes n times. It needs to be a self contained example. I should be able to set the value of n and then run your loop on my computer.

```
n = 14
for (i in 1:n)
{
  print(i)
}
```

Q8 (4 pts.): Provide the code you used to create the `n`, `vec_1`, and the loop. As always, it should run as a stand-alone example in a fresh R session on my computer.

```
n=17
vec_1 = sample(10,n,replace = TRUE)
head(vec_1)
for (i in 1:n) {
  print(
    paste0("The element of vec_1 at index ",i, " is ",vec_1[i]))
}
```

Q9 (10 pts.): Provide the code you used to build your function.

To receive full credit your code must run without error on a new R session and produce output similar to the examples given in the instructions.

```
create_and_print_vec = function(n, min = 1, max =10)
{
  vec_4 = sample(min:max,n,replace = FALSE)
  for (i in 1:n) {
    print(
      paste0("The element of vec_4 at index ",i, " is ",vec_4[i]))
  }
}
create_and_print_vec(10, min = 100, max = 2000)
```