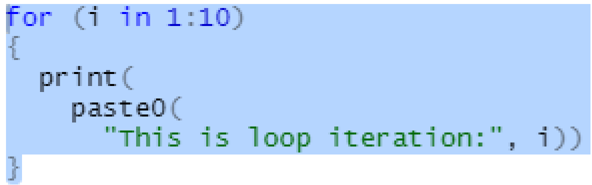
Keegan Moynahan

Lab 2

Work on the lab with Steph

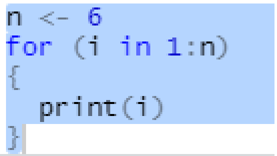
1) vec_2 = vec_1 == 3
2) One reason why visual selection would be bad because then you couldn't store the values as true or false which are easier to read visually then trying to go through numbers to find a certain value. Another reason is if the data was changed then you would have to continue visually inspecting when with the Boolean function you can just plug it into new data.
3) I didn't get the same number of 3 each time because it is randomly selecting the values and there might not be a 3 every time running it.
4) A logical text is the safest way because it will select the correct values (3) and tell you which ones aren't 3. Each time running the vector you get different values and logical test is the quickest and most efficient way to pick the 3's out, you can also just plug the vector into the logical function. This tool is extremely helpful when using the function to select values greater than something.
5) By hand sub setting is bad because the vectors will change when dealing with random sampling. Also, if the data is large going through the data your bound to miss values that should be true or mark values as true that are false. Like I explained above using this tool would be useful for selecting data greater than a value. This tool is extremely reusable.

6)



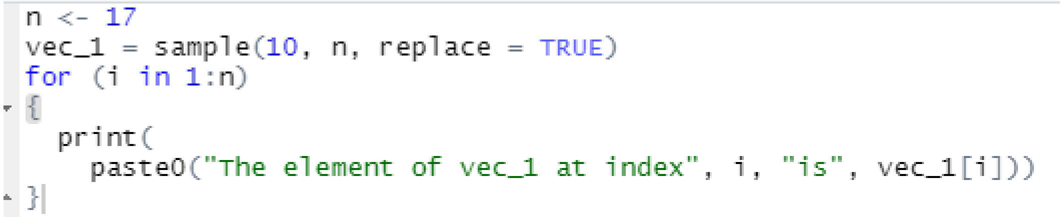
```
for (i in 1:10)
{
  print(
    paste0(
      "This is loop iteration:", i))
}
```

7)



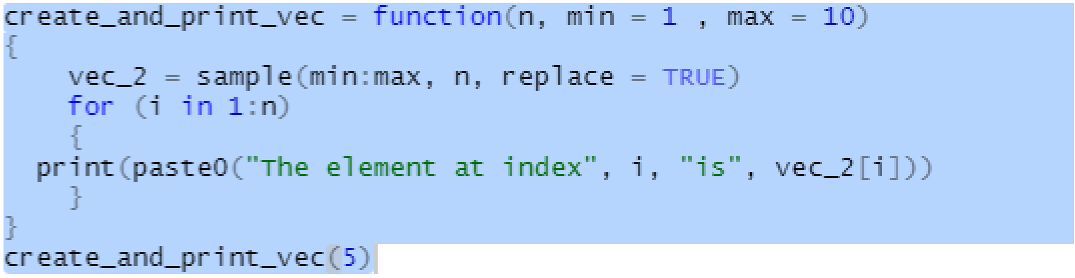
```
n <- 6
for (i in 1:n)
{
  print(i)
}
```

8)



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

9)



```
create_and_print_vec = function(n, min = 1 , max = 10)
{
    vec_2 = sample(min:max, n, replace = TRUE)
    for (i in 1:n)
    {
  print(paste0("The element at index", i, "is", vec_2[i]))
    }
}
create_and_print_vec(5)
```

Work on the lab with Steph