Today, we are going to cover repeat loops.

To find the average of math scores, we may use the following repeat loop.

sum <- 0  
idx <- 1  
repeat {  
 sum <- sum + StudentsPerformance[idx, "MathScore" ]  
 idx <- idx + 1  
 if (idx > nrow(StudentsPerformance)){  
 break  
 }  
}  
avg <- sum/nrow(StudentsPerformance)  
print(avg)

It is a very special loop:

* There is no logical condition to check at the beginning of the loop to terminate it.
* We have to write some if conditions in the body of the ‘repeat’ loop to check and terminate it explicitly using **break** statement.

The R codes work as follows:

* Initialize the sum to 0 and index to 1 that holds the row number
* First iteration of the loop:
  + Add the 1st row Math score to sum since idx = 1.
  + Update the row number by adding 1; i.e. idx = 1+1 =2
  + Check if condition “idx > nrow(StudentsPerformance)”, which is 2> 1000; since it is FALSE, the body of if is not executed
* Second iteration of the loop:
  + Add the 2nd row Math score to sum since idx = 2.
  + Update the row number by adding 1; i.e. idx = 2+1 =3
  + Check if condition “idx > nrow(StudentsPerformance)”, which is 3> 1000; since it is FALSE, the body of if is not executed
* …
* 1000th iteration of the loop:
  + Add the 1000th row Math score to sum since idx = 1000.
  + Update the row number by adding 1; i.e. idx = 1000+1 =1001
  + Check if condition “idx > nrow(StudentsPerformance)”, which is 1001> 1000; since it is TRUE, the body of if, break, is executed. The loop is terminated.
* Compute the average by dividing the sum of math scores by the number of rows outside of the loop.

Let’s look at the result by running the R codes. The average is 66.

We learned that the “break” statement terminates the loop. There is another statement in loops in R. It is the “next” statement. The “next” statement **only terminates the current iteration** and then goes to the next iteration.

Your manager asks you to find the average math scores for those students whose math scores is higher than 80 from scratch. We can use the following R codes.

sum80 <- 0  
count80 <- 0  
for (idx in c(1:nrow(StudentsPerformance))){  
 if (StudentsPerformance[idx, "MathScore" ] <= 80)  
 next  
 sum80 <- sum80 + StudentsPerformance[idx, "MathScore" ]  
 count80 <- count80 + 1  
}  
avg80 <- sum80/count80  
print(avg80)

The R codes work as follows:

* Initialize the sum80 to 0 and count80 to 0 that holds the row numbers processed
* First iteration of the loop, idx = 1:
  + Compare 1st row of math score to 80
  + If it is less than or equal to 80, skip the current iteration(idx =1), then go to the 2nd iteration with idx =2
  + If it is larger than 80, then run the current iteration
    - Add the corresponding math score to the sum of 80
    - Update the number of records whose math is higher than 80 by increasing by 1
* Second iteration of the loop, idx = 2:
  + Compare 2nd row of math score to 80
  + If it is less than or equal to 80, skip the current iteration(idx = 2), then go to the 3rd iteration with idx = 3
  + If it is larger than 80, then run the current iteration
    - Add the corresponding math score to the sum of 80
    - Update the number of records whose math is higher than 80 by increasing by 1
* …
* 1000th iteration of the loop, idx = 1000
  + Compare 1000th row of math score to 80
  + If it is less than or equal to 80, skip the current iteration(idx = 1000), then it tries to go to next iteration with idx = 1001 which is not in the range of the colon operator. The loop is terminated.

Let’s look at the result by running the codes. The average math scores for those students whose math scores is higher than 80 is 87.7.

Note that there are other solutions. For example, we can use the following R codes:

sum80 <- 0  
count80 <- 0  
for (idx in c(1:nrow(StudentsPerformance))){  
 if (StudentsPerformance[idx, "MathScore" ] > 80){  
 sum80 <- sum80 + StudentsPerformance[idx, "MathScore" ]  
 count80 <- count80 + 1  
 }  
}  
avg80 <- sum80/count80  
print(avg80)

We run it and get the same result that is 87.7.

The three loops in R, **for**, **while**, and **repeat** are equivalent to each other. When we write the R codes, we often use one of them, for example, **for** loops, which is easier since we don’t need to update the index explicitly. But we have to do it for the other two. It is our obligation to terminate the **repeat** loop using the if condition or **break** statement.