Homework1

Olivia Triska, Jared ‘Jay’ Klein

9/3/2021

1. Use and show R code to produce all even integers greater than 7 but less than 101.

N<-7:101  
N<-N[N %% 2 == 0]  
print(N)

## [1] 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44  
## [20] 46 48 50 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82  
## [39] 84 86 88 90 92 94 96 98 100

1. Use and show R code to find the mean of all positive integers less than 100.

X<-0:99  
mean(X)

## [1] 49.5

1. Write a function (using r code and structure demonstrated in class) to calculate a z score for a given observed value, a mean, and a standard deviation value. And then use your function to find a z score for the following problem. (Research the internet to find the formula used to calculate a z score)Observed value = 25.77, mean = 23.54, standard deviation = 2.442

get\_dat\_z\_score\_bb <- function(obv, mean\_num, std\_dev\_num, round=FALSE){  
 z\_score = (obv - mean\_num)/std\_dev\_num  
   
 if (round == TRUE) {  
 rounded\_z\_score <- round(z\_score, digits = 1)  
 return(rounded\_z\_score)  
 }  
   
 return(z\_score)  
}  
  
print(get\_dat\_z\_score\_bb(25.77, 23.54, 2.442))

## [1] 0.9131859

1. Use and show R code to round your z score answer in problem 3 to the nearest tenth

print(get\_dat\_z\_score\_bb(25.77, 23.54, 2.442, round = TRUE))

## [1] 0.9

1. Write a function (using r code and the structure demonstrated in class) to calculate the natural log of a number multiplied by the common log of the same number divided by the cube root of a given prime number.Use your function to find the answer if the number to be used for both log expressions is 32 and the given prime number is 11.

calc\_func <- function(num, prime\_num){  
 result = (log(num) \* log10(num)) / (prime\_num^(1/3))  
 return(result)  
}  
  
print(calc\_func(32, 11))

## [1] 2.345548

1. Use and show R code to round your answer in problem 5 to the nearest hundredth.

print(round(calc\_func(32, 11), digits = 2))

## [1] 2.35

1. Write a function (using r code and the structure demonstrated in class) that will create the first10 squares of positive odd integers. That is your function will give you the following output. 1, 9, 25, 49, …, 361 when you input values 1 to 10.

first\_10\_sqr\_nxt <- function(min\_value, max\_value){  
 values <- min\_value:max\_value  
 new\_value\_vector <- values[values %% 2 != 0]  
 new\_value\_vector <- new\_value\_vector^2  
 return(new\_value\_vector)  
}  
  
first\_10\_sqr\_nxt(1, 10)

## [1] 1 9 25 49 81