Homework03

Peyton Hall

02/01/2024

# What does the following function return? Why?   
# (Write comments in your RMarkdown file with a   
# couple of sentences to explain why you obtained such an output).  
  
F1<-function(x=2, y=0) {  
 x+y  
}  
F1(3,4)

## [1] 7

# the function takes two parameters, x and y, and it calculates   
# the sum of them and returns the result.   
# F1(3,4) overrides the default values, x and y, and it   
# calculates the sum of 3 and 4.

# What does the following function return? Why? (Write comments   
# in your RMarkdown file to explain why your output is like that)  
l<-function (x){  
result<-x+1  
return(result)  
}  
m<-function(){  
l<-function(x) {  
result<-x\*2  
return(result)  
}  
l(10)  
}  
m()

## [1] 20

# The first function is defined with one parameter, x. In it, it # calculates the sum of x and 1 and returns the result.  
# The second function, m, is defined. Inside m, there is a local # function, l, defined again, which has a different   
# implementation than the first l. The local l function takes   
# a parameter, x, and calculates result as the product of x and   
# 2. Then, it returns the result.  
# Finally, the m function calls the local l function, with 10 as   
# the argument. Therefore, m() returns the result, of calling the # local l(10).

# Create a function in R to calculate the T statistic for the   
# one-sample T-test  
# T = (𝑥̅ - 𝜇)/(s/SQRTn)  
# Where  
# 𝑥̅ = sample mean, ("x bar")  
# 𝜇 = population mean,  
# s = sample standard deviation  
# n = sample size.  
# Suppose the input arguments are 𝒙̅, 𝝁, 𝒔 and 𝒏  
# Call your function with the inputs: 𝒙̅=2.1, 𝝁=1, 𝒔=0.5, n=50  
  
# Function to calculate T statistic for one-sample T-test  
calculate\_T\_statistic <- function(x\_bar, mu, s, n) {  
 t\_statistic <- (x\_bar - mu) / (s / sqrt(n))  
 return(t\_statistic)  
}  
  
# Input arguments  
x\_bar <- 2.1  
mu <- 1  
s <- 0.5  
n <- 50  
  
# Call the function  
result <- calculate\_T\_statistic(x\_bar, mu, s, n)  
  
# Print the result  
cat("T statistic:", result, "\n")

## T statistic: 15.55635

# Your collaborator tells you that you can use the length of the # hindfoot to calculate brain volume.  
# Apparently, the hindfoot of these creatures is equal to the   
# diameter of their skulls. Write a function that will calculate # the volume of the animals’ skulls. The formula for the volume   
# of a sphere is (4∗𝜋∗𝑟^3)/3 , where r is the radius of the   
# skull.  
# Include one input: d being the diameter of the skull.  
# Call the function with the inputs: 𝒅 = 𝟓, where 𝑑 is the  
# diameter (hindfoot) of a skull.  
  
# Function to calculate volume of animal's skull  
calculate\_skull\_volume <- function(d) {  
 # Calculate the radius from the diameter  
 r <- d / 2  
   
 # Calculate the volume using the formula for the volume of a sphere  
 volume <- (4 \* pi \* r^3) / 3  
   
 return(volume)  
}  
  
# Input diameter  
d <- 5 # replace with the actual hindfoot diameter  
  
# Call the function  
skull\_volume <- calculate\_skull\_volume(d)  
  
# Print the result  
cat("Volume of the animal's skull:", skull\_volume, "\n")

## Volume of the animal's skull: 65.44985

# Create a function to find the center of mass for two masses,   
# with four parameters (input or arguments) being m1, m2, x1 and # x2, where m1 and m2 are the mass of the two and x1 and x2  
# are the locations of the two masses. The following figure   
# showed the calculation of the center of  
# two masses  
  
# x\_cm = (m\_1\*x\_1 + m\_2\*x\_2)/(m\_1 + m\_2)  
  
# Call the function with the inputs: 𝑚\_1 = 2, 𝑚\_2 = 5, 𝑥\_1 = 3, 𝑎𝑛𝑑 𝑥\_2 = 10  
  
# Function to calculate the center of mass for two masses  
calculate\_center\_of\_mass <- function(m1, m2, x1, x2) {  
 # Calculate the center of mass using the formula  
 x\_cm <- (m1 \* x1 + m2 \* x2) / (m1 + m2)  
   
 return(x\_cm)  
}  
  
# Input parameters  
m1 <- 2  
m2 <- 5  
x1 <- 3  
x2 <- 10  
  
# Call the function  
center\_of\_mass <- calculate\_center\_of\_mass(m1, m2, x1, x2)  
  
# Print the result  
cat("Center of mass:", center\_of\_mass, "\n")

## Center of mass: 8