

SESSION 1: INTRODUCTION

Assignment 2

1. How many ways are there to call a function in R?

Answer: Function in R can be called in 3 ways

1. **Function call by Name:** In this method, the argument name is been used along with value while calling the function.
2. **Function call by Position:** In this method, only the value are been passed in the argument, but R assigns the value to the function variable based on their positions.
3. **Function without arguments:** Here the function doesn't have any arguments

Example for function calling by name and by position

```
func1 <- function (left, right){  
  print (paste("value of left is ", as.character(left) ))  
  print (paste("value of right is ", as.character(right) ))  
}
```

Function call by name

```
func1 (left=24, right=12)  
> # Function call by name  
> func1(left=24,right=12)  
[1] "value of left is 24"  
[1] "value of right is 12"
```

Function call by name

```
func1 (24, 12)  
> # Function call by position  
> func1(24,12)  
[1] "value of left is 24"  
[1] "value of right is 12"
```

Example for function calling without arguments

```
func2 <- function(){  
  sumVal = 12 + 13  
  print (paste("Additon of 12 and 13 is ", as.character(sumVal) ))  
}  
func2 <- function(){  
+   sumVal <- 12 + 13  
+   print (paste("Additon of 12 and 13 is ", as.character(sumVal) ))  
+ }  
> # Function call without arguments  
> func2()  
[1] "Additon of 12 and 13 is 25"
```

2. Is the below statement true?

- The lazy evaluation of a function means, the argument is evaluated only if it is evaluated only if it is used inside the body of the function. **TRUE**

Lazy evaluation means an evaluation strategy which delays the evaluation of an expression until its value is needed and thus avoids repetitive evaluations

Example for Lazy evaluation

```
x<-function(a,b){  
  print (a)  
  print (b)  
}
```

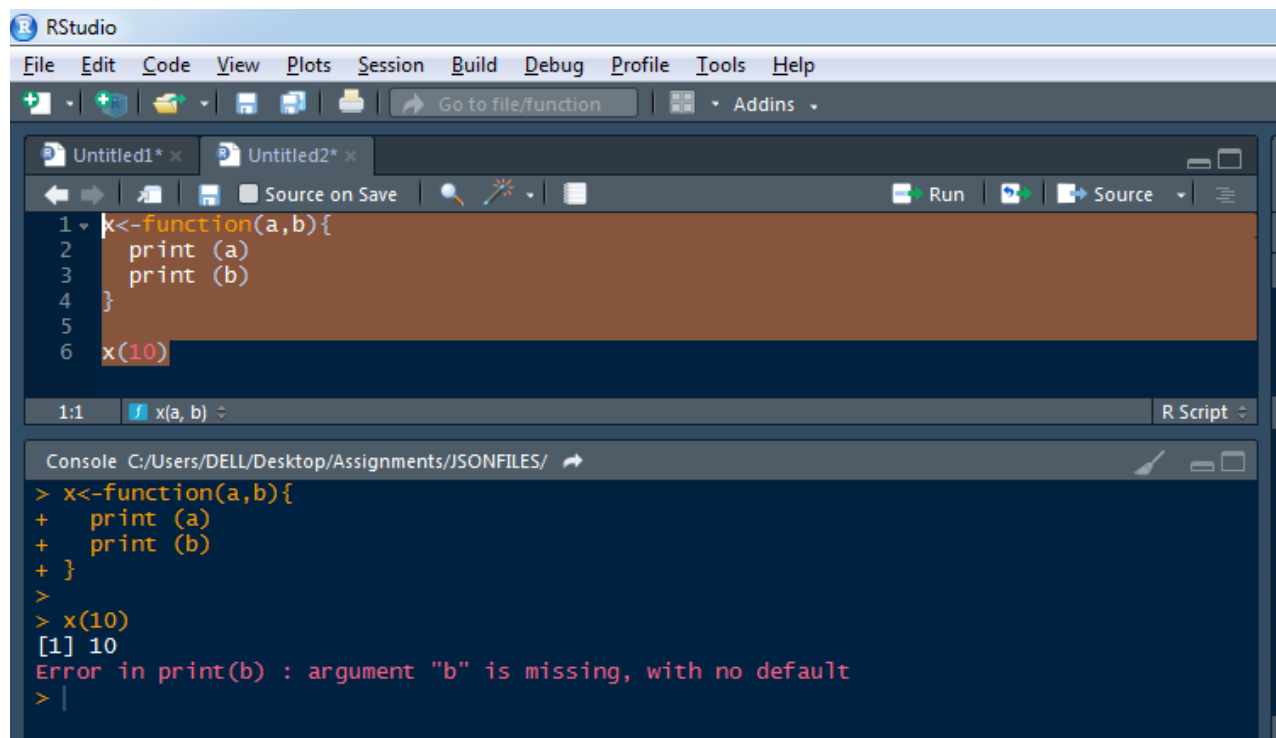
x(10)

Here only one value for a is been passed to the function calling by position,

When the function is called, first print statement is printing value of a

When the second print statement is trying to print value of b which is missing then we are getting error.

Means, the error is not thrown in the beginning of the function, rather, at run-time when the error raises.



The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 x<-function(a,b){  
2   print (a)  
3   print (b)  
4 }  
5  
6 x(10)
```

The console output shows the execution of the function:

```
> x<-function(a,b){  
+   print (a)  
+   print (b)  
+ }  
>  
> x(10)  
[1] 10  
Error in print(b) : argument "b" is missing, with no default  
> |
```

3. Mention true or false for below statements:

- a. Insights driven from descriptive analytics is not meaningful. **FALSE**
- b. The number of values in each Elements of a list, should be equal. **FALSE**
- c. The datasets are not stored in memory of the computer using R. **FALSE**
- d. Data frames and matrices are two dimensional however the array is multidimensional. **TRUE**