Quiz-3.R

ASTHAGINA DELIA P

2024-09-19

#CALL A FUNCTION  
my\_function = function(){  
 print("say the name")  
}  
 my\_function()

## [1] "say the name"

#ARGUMENTS IN FUNCTION  
my\_function= function(fname) {  
 paste(fname, "lee")  
}  
my\_function("seokmin")

## [1] "seokmin lee"

my\_function("jihoon")

## [1] "jihoon lee"

my\_function("chan")

## [1] "chan lee"

#NUMBER OF ARGUMENTS IN FUNCTION  
my\_function= function(fname, lname) {  
 paste(fname, lname)  
}  
my\_function("Jeon", "Wonwoo")

## [1] "Jeon Wonwoo"

#DEFAULT PARAMETER VALUE   
my\_function= function(country = "United Kingdom") {  
 paste("i am from", country)  
}  
my\_function("England")

## [1] "i am from England"

my\_function("Scotland")

## [1] "i am from Scotland"

my\_function()

## [1] "i am from United Kingdom"

my\_function("North Ireland")

## [1] "i am from North Ireland"

my\_function("Wales")

## [1] "i am from Wales"

#RETURN VALUE IN FUNCTION  
my\_function= function(x) {  
 return(19\*x)  
}  
print(my\_function(2))

## [1] 38

print(my\_function(7))

## [1] 133

print(my\_function(0))

## [1] 0

#NESTED IN FUNCTION  
#1  
nested\_function = function(x, y){  
 a = x + y  
 return(a)  
}  
nested\_function(nested\_function(5,6), nested\_function(8,7))

## [1] 26

#2  
outer\_func = function(x){  
 inner\_func= function(y){  
 a= x+y  
 return(a)  
 }  
 return(inner\_func)  
}  
output= outer\_func(6)  
output(8)

## [1] 14

#RECURSION IN FUNCTION  
tri\_recursion = function(k) {  
 if(k>1) {  
 result = k+tri\_recursion(k-1)  
 print(result)  
 } else {  
 result = 1  
 return(result)  
 }  
}  
tri\_recursion(9)

## [1] 3  
## [1] 6  
## [1] 10  
## [1] 15  
## [1] 21  
## [1] 28  
## [1] 36  
## [1] 45

#GLOBAL VARIABLE  
#1  
txt = "very demure"  
my\_function = function() {  
 paste("She is", txt)  
}  
my\_function()

## [1] "She is very demure"

#2  
txt = "SEVENTEEN"  
my\_function= function() {  
 txt= "funny"  
 paste("Going Seventeen is", txt)  
}  
my\_function()

## [1] "Going Seventeen is funny"

txt

## [1] "SEVENTEEN"

#THE GLOBAL ASSIGNMENT OPERATOR IN FUNCTION  
#1  
my\_function = function() {  
 txt <<- "Gorgeous"  
 paste("Niki Zefanya is", txt)  
}  
my\_function()

## [1] "Niki Zefanya is Gorgeous"

print(txt)

## [1] "Gorgeous"

#2  
txt= "coquette"  
my\_function= function() {  
 txt<="pretty"  
 paste("you are so", txt)  
}  
my\_function()

## [1] "you are so coquette"

paste("you are so", txt)

## [1] "you are so coquette"

#MACAM-MACAM FUNCITON  
#built-in  
#1  
rata\_rata= median(c(3, 4, 7, 10))  
print(rata\_rata)

## [1] 5.5

#2  
hitungRataRata= function(angka) {  
 rata\_rata = mean(angka)  
 return(rata\_rata)  
}  
hitungRataRata(c(5, 7, 9, 15))

## [1] 9

#3  
jumlah = function(...) {  
 sum(...)  
}  
jumlah(123, 223, 323, 423, 523)

## [1] 1615

#USER DEFINED (FUNGSI CUSTOM)  
hitungLuasPersegiPanjang = function(panjang, lebar) {  
 luas = panjang \* lebar  
 return(luas)  
}  
hitungLuasPersegiPanjang(12,10)

## [1] 120

#PACKAGE & LIBRARY  
#modul eksternal  
#contoh kasus 1  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

data = data.frame(nama = c("Wonu", "Dino", "Igyu"), nilai = c(78, 86, 92))  
data\_filtered = data %>% filter(nilai > 78)  
print(data\_filtered)

## nama nilai  
## 1 Dino 86  
## 2 Igyu 92

#conoth kasus 2  
library(ggplot2)  
data\_penjualan = data.frame(  
 nama = c("Senin", "Selasa", "Rabu", "Jumat"),  
 penjualan = c(8000, 2000, 6800, 10000)  
)  
  
ggplot(data\_penjualan, aes(x = nama, y = penjualan)) +  
 geom\_bar(stat = "identity", fill = "pink") +   
 ggtitle("penjualan warung 17")

