Tugas Modul 5

Nur Rosydatun Nafiah

2022-09-26

library(dslabs)  
data("murders")

## Nomor 1

1. Fungsi nchar dapat digunakan untuk menghitung jumlah karakter dari suatu vektor karakter.Buatlah satu baris kode yang akan menyimpan hasil komputasi pada variabel ‘new\_names’ dan berisi singkatan nama negara ketika jumlah karakternya lebih dari 8 karakter.

Jawab :

new\_names <- ifelse(nchar(murders$state) > 8, murders$abb, murders$state)  
new\_names

## [1] "Alabama" "Alaska" "Arizona" "Arkansas" "CA" "Colorado"  
## [7] "CT" "Delaware" "DC" "Florida" "Georgia" "Hawaii"   
## [13] "Idaho" "Illinois" "Indiana" "Iowa" "Kansas" "Kentucky"  
## [19] "LA" "Maine" "Maryland" "MA" "Michigan" "MN"   
## [25] "MS" "Missouri" "Montana" "Nebraska" "Nevada" "NH"   
## [31] "NJ" "NM" "New York" "NC" "ND" "Ohio"   
## [37] "Oklahoma" "Oregon" "PA" "RI" "SC" "SD"   
## [43] "TN" "Texas" "Utah" "Vermont" "Virginia" "WA"   
## [49] "WV" "WI" "Wyoming"

## 

## Nomor 2

1. Buat fungsi sum\_n yang dapat digunakan untuk menghitung jumlah bilangan bulat dari 1 hingga n. Gunakan pula fungsi ini untuk menentukan jumlah bilangan bulat dari 1 hingga 5.000.

Jawab :

sum\_n <- function(n) {   
 x = 1:n  
 sum(x)  
}  
sum\_n(5000)

## [1] 12502500

## Nomor 3

1. Buat fungsi compute\_s\_n yang dapat digunakan untuk menghitung jumlah Sn = 1^2 + 2^2 + 3^2 + . . . n^2. Tampilkan hasil penjumlahan ketika n = 10.

Jawab :

n = 10  
compute\_s\_n <- function(n){  
 x = 1:n  
 sum(x^2)  
}  
compute\_s\_n(n)

## [1] 385

## 

## Nomor 4

1. Buat vektor numerik kosong dengan nama: s\_n dengan ukuran:25 menggunakan s\_n <- vector (“numeric”, 25).Simpan di hasil komputasi S1, S2,. . . S25 menggunakan FOR-LOOP.

Jawab :

s\_n <- vector ("numeric", 25)  
for(i in 1:length(s\_n)){  
 s\_n[i] <- compute\_s\_n(i)  
}  
s\_n

## [1] 1 5 14 30 55 91 140 204 285 385 506 650 819 1015 1240  
## [16] 1496 1785 2109 2470 2870 3311 3795 4324 4900 5525

## 

## Nomor 5

1. Ulangi langkah pada soal no. 4 dan gunakan fugsi sapply.

Jawab :

n <- 1:25  
sapply(n, compute\_s\_n)

## [1] 1 5 14 30 55 91 140 204 285 385 506 650 819 1015 1240  
## [16] 1496 1785 2109 2470 2870 3311 3795 4324 4900 5525