Tugas 3_Modul7

Farhan Harvito

2022-11-02

1. Menggunakan as_tibble untuk mengkonversi tabel dataset “US murders” dalam bentuk tibble dan menyimpannya dalam objek baru bernama ‘murders_tibble’

```
library(dslabs)
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.2

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✔ ggplot2 3.3.6      ✔ purrr   0.3.4
## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.0      ✔ stringr 1.4.1
## ✔ readr   2.1.2      ✔ forcats 0.5.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()

data(murders)
murders_tibble <- murders%>%as_tibble()
```

2. Menggunakan fungsi group_by untuk mengkonversi dataset “US murders” menjadi sebuah tibble yang dikelompokkan berdasarkan ‘region’

```
as_tibble(murders) %>% group_by(region)

## # A tibble: 51 × 5
## # Groups:   region [4]
##    state                abb   region    population total
##    <chr>                <chr> <fct>          <dbl> <dbl>
##  1 Alabama              AL    South        4779736   135
##  2 Alaska               AK    West          710231    19
##  3 Arizona              AZ    West         6392017   232
##  4 Arkansas             AR    South        2915918    93
##  5 California           CA    West        37253956  1257
##  6 Colorado             CO    West         5029196    65
##  7 Connecticut          CT    Northeast    3574097    97
##  8 Delaware             DE    South         897934    38
##  9 District of Columbia DC    South         601723    99
## 10 Florida              FL    South       19687653   669
## # … with 41 more rows
```

3. Menggunakan operator pipe dan dot operator

```
library(dslabs)
library(dplyr)
data(murders)
murders %>%
     pull(population) %>%
     log %>%
     mean %>%
     exp
```

```
## [1] 3675209
```

```
mapping <- function(n){
  baris<-1:n
  result<-0
  for(i in baris){
    result <- result + i
    result2 <- result^2
  }
  tibble(
    n,
    s_n = result,
    s_n_2 = result2
  )
}

x = 1:100
map_df(x,mapping)
```

```
## # A tibble: 100 × 3
##        n   s_n s_n_2
##    <int> <dbl> <dbl>
##  1     1     1     1
##  2     2     3     9
##  3     3     6    36
##  4     4    10   100
##  5     5    15   225
##  6     6    21   441
##  7     7    28   784
##  8     8    36  1296
##  9     9    45  2025
## 10    10    55  3025
## # … with 90 more rows
```