

modul 7

Eva

11/7/2021

Latihan

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

```
library(dslabs)
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5      v purrr  0.3.4
## v tibble  3.1.4      v dplyr  1.0.7
## v tidyr   1.1.4      v stringr 1.4.0
## v readr   2.0.2      v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

data(murders)
as_tibble(murders) %>% class()

## [1] "tbl_df"      "tbl"        "data.frame"

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

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

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

## # A tibble: 51 x 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. Tulis script tidyverse yang menghasilkan output yang sama dengan perintah berikut: `exp(mean(log(murders$population)))`. Gunakan operator pipe sehingga setiap fungsi dapat dipanggil tanpa menambahkan argumen. Gunakan dot operator untuk mengakses populasi.

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

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

4. Gunakan `map_df` untuk membuat data frame yang terdiri dari tiga kolom: 'n', 's_n', dan 's_n_2'. Kolom pertama harus berisi angka 1 hingga 100. Kolom kedua dan ketiga masing-masing harus berisi penjumlahan 1 hingga n, dimana n menyatakan jumlah baris.

```
library(purrr)
compute_s_n <- function(n){
  x <- 1:n
  sum(x)
}
n <- 1:100
s_n <- sapply(n, compute_s_n)
compute_s_n <- function(n){
  x <- 1:n
  tibble(sum = sum(x))
}
s_n <- map_df(n, compute_s_n)
as_tibble(s_n)
```

```
## # A tibble: 100 x 1
##       sum
##   <int>
## 1     1
## 2     3
## 3     6
## 4    10
## 5    15
## 6    21
## 7    28
## 8    36
## 9    45
## 10   55
## # ... with 90 more rows
```