

Latihan4_123190125

Harya Anggitama

11/3/2021

```
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   1.1.3      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()

library(dslabs)
data(murders)
```

Nomor 1

Menggunakan `as_tibble` untuk mengonversi data set `us_murders`

```
murders_tibble <- as_tibble(murders)
head(murders_tibble)

## # A tibble: 6 x 5
##   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
```

Nomor 2

Menggunakan fungsi `group_by` untuk mengonversi data set `us_murders` menjadi sebuah tibble yang dikelompokkan berdasarkan region:

```
murders_tibble_region <- murders %>%
  group_by(region) %>%
  as_tibble()

murders_tibble_region
```

```
## # A tibble: 51 x 5
##   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
```

Nomor 3

Menulis script tidyverse yang menghasilkan output yang sama dengan perintah berikut:

```
exp(mean(log(murders$population)))
```

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

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

Nomor 4

Menggunakan fungsi `map_df` untuk membuat data frame yang terdiri dari tiga kolom: 'n', 's_n', dan 's_n_2'

```
compute_s_n <- function(n) {
  x <- 1:n
  for (row in x) {

  }
  tibble(n = x)
}

s_n <- map_df(100, compute_s_n)

s_n
```

```
## # A tibble: 100 x 1
##       n
##   <int>
## 1     1
## 2     2
## 3     3
## 4     4
## 5     5
## 6     6
## 7     7
## 8     8
## 9     9
## 10    10
## # ... with 90 more rows
```