

Rworksheet_Mabalina#2

Kristian B. Mabalina

2024-09-25

#1

#a

```
seq(-5, 5)
```

```
## [1] -5 -4 -3 -2 -1 0 1 2 3 4 5
```

```
x <- (1:7)
```

#b

```
x <- (1:7)
```

#2

```
seq(1, 3, by=0.2)
```

```
## [1] 1.0 1.2 1.4 1.6 1.8 2.0 2.2 2.4 2.6 2.8 3.0
```

#3

```
workers <- c(34, 28, 22, 36, 27, 18, 52, 39, 42, 29, 35, 31, 27,  
22, 37, 34, 19, 20, 57, 49, 50, 37, 46, 25, 17, 37, 43, 53, 41, 51, 35,  
24, 33, 41, 53, 40, 18, 44, 38, 41, 48, 27, 39, 19, 30, 61, 54, 58, 26,  
18)
```

#a

```
Workers <- c(34, 28, 22, 36, 27, 18, 52, 39, 42, 29, 35, 31, 27,  
22, 37, 34, 19, 20, 57, 49, 50, 37, 46, 25, 17, 37, 43, 53, 41, 51, 35,  
24, 33, 41, 53, 40, 18, 44, 38, 41, 48, 27, 39, 19, 30, 61, 54, 58, 26,  
18)
```

```
work <- Workers
```

```
work
```

```
## [1] 34 28 22 36 27 18 52 39 42 29 35 31 27 22 37 34 19 20 57 49 50 37 46 25 17
```

```
## [26] 37 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
```

#b

```
work <- c(2,4)
```

#c

```
work <- c(-4,12)
```

#4

```
x <- c("first"=3, "second"=0, "third"=9)
```

#a

```
selected_elements <- c("first", "third")
```

```
#b
```

```
selected_elements
```

```
## [1] "first" "third"
```

```
#5
```

```
seq(-3,2)
```

```
## [1] -3 -2 -1  0  1  2
```

```
#a
```

```
x[2] <-c(0)
```

```
#b
```

```
#x[2] <-c(0)
```

```
#
```

```
#6 #a
```

```
diesel_fuel <- data.frame(  
Month = c("Jan", "Feb", "March", "Apr", "May", "June"),  
Php = c(52.50, 57.25, 65.00, 60.00, 74.25, 54.00),  
liter = c(25, 30, 40, 50, 10, 45)  
)
```

```
#b
```

```
weighted.mean <-c(diesel_fuel$liter, diesel_fuel$purchase)
```

```
#7
```

```
rivers
```

```
## [1] 735 320 325 392 524 450 1459 135 465 600 330 336 280 315 870  
## [16] 906 202 329 290 1000 600 505 1450 840 1243 890 350 407 286 280  
## [31] 525 720 390 250 327 230 265 850 210 630 260 230 360 730 600  
## [46] 306 390 420 291 710 340 217 281 352 259 250 470 680 570 350  
## [61] 300 560 900 625 332 2348 1171 3710 2315 2533 780 280 410 460 260  
## [76] 255 431 350 760 618 338 981 1306 500 696 605 250 411 1054 735  
## [91] 233 435 490 310 460 383 375 1270 545 445 1885 380 300 380 377  
## [106] 425 276 210 800 420 350 360 538 1100 1205 314 237 610 360 540  
## [121] 1038 424 310 300 444 301 268 620 215 652 900 525 246 360 529  
## [136] 500 720 270 430 671 1770
```

```
#a
```

```
data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers), var(rivers),  
sd(rivers), min(rivers), max(rivers))  
print(data)
```

```
## [1] 141.0000 83357.0000 591.1844 425.0000 243908.4086 493.8708  
## [7] 135.0000 3710.0000
```

```
#b
```

```
#[1] 141.0000 83357.0000 591.1844 425.0000 243908.4086 493.8708  
#[7] 135.0000 3710.0000
```

#8-a

```
power_ranking <- 1:25
celebrity_name <- c("Tom Cruise", "Rolling Stones", "Oprah Winfrey", "U2", "Tiger Woods", "Steven Spielberg", "Howard Stern", "50 Cent", "Cast of the Sopranos", "Dan Brown", "Bruce Springsteen", "Donald Trump", "Muhammad Ali", "Paul McCartney", "George Lucas", "Elton John", "David Letterman", "Phil Mickelson", "J.K. Rowling", "Bradd Pitt", "Peter Jackson", "Dr. Phil McGraw", "Jay Leno", "Celine Dion", "Kobe Bryant")
pay <- c(67, 90, 225, 110, 90, 332, 302, 41, 52, 88, 55, 44, 55, 40, 233, 34, 40, 47, 75, 25, 39, 45, 31)
celebrity_data <- data.frame(PowerRanking = power_ranking, CelebrityName = celebrity_name, Pay = pay)
celebrity_data
```

##	PowerRanking	CelebrityName	Pay
## 1	1	Tom Cruise	67
## 2	2	Rolling Stones	90
## 3	3	Oprah Winfrey	225
## 4	4	U2	110
## 5	5	Tiger Woods	90
## 6	6	Steven Spielberg	332
## 7	7	Howard Stern	302
## 8	8	50 Cent	41
## 9	9	Cast of the Sopranos	52
## 10	10	Dan Brown	88
## 11	11	Bruce Springsteen	55
## 12	12	Donald Trump	44
## 13	13	Muhammad Ali	55
## 14	14	Paul McCartney	40
## 15	15	George Lucas	233
## 16	16	Elton John	34
## 17	17	David Letterman	40
## 18	18	Phil Mickelson	47
## 19	19	J.K. Rowling	75
## 20	20	Bradd Pitt	25
## 21	21	Peter Jackson	39
## 22	22	Dr. Phil McGraw	45
## 23	23	Jay Leno	32
## 24	24	Celine Dion	40
## 25	25	Kobe Bryant	31

#b

```
celebrity_data[celebrity_data$CelebrityName == "J.K. Rowling", "PowerRanking"] <- 15
celebrity_data[celebrity_data$CelebrityName == "J.K. Rowling", "Pay"] <- 90
```

celebrity_data

##	PowerRanking	CelebrityName	Pay
## 1	1	Tom Cruise	67
## 2	2	Rolling Stones	90
## 3	3	Oprah Winfrey	225
## 4	4	U2	110
## 5	5	Tiger Woods	90
## 6	6	Steven Spielberg	332
## 7	7	Howard Stern	302
## 8	8	50 Cent	41
## 9	9	Cast of the Sopranos	52
## 10	10	Dan Brown	88
## 11	11	Bruce Springsteen	55
## 12	12	Donald Trump	44
## 13	13	Muhammad Ali	55

```
## 14      14      Paul McCartney 40
## 15      15      George Lucas 233
## 16      16      Elton John 34
## 17      17      David Letterman 40
## 18      18      Phil Mickelson 47
## 19      15      J.K. Rowling 90
## 20      20      Bradd Pitt 25
## 21      21      Peter Jackson 39
## 22      22      Dr. Phil McGraw 45
## 23      23      Jay Leno 32
## 24      24      Celine Dion 40
## 25      25      Kobe Bryant 31
```

```
#c
```

```
write.csv(celebrity_data, file = "PowerRanking.csv", row.names = FALSE)
```

```
celebrity_data_imported <- read.csv("PowerRanking.csv")
```

```
celebrity_data_imported
```

```
##      PowerRanking      CelebrityName Pay
## 1          1      Tom Cruise 67
## 2          2      Rolling Stones 90
## 3          3      Oprah Winfrey 225
## 4          4              U2 110
## 5          5      Tiger Woods 90
## 6          6      Steven Spielberg 332
## 7          7      Howard Stern 302
## 8          8      50 Cent 41
## 9          9      Cast of the Sopranos 52
## 10         10      Dan Brown 88
## 11         11      Bruce Springsteen 55
## 12         12      Donald Trump 44
## 13         13      Muhammad Ali 55
## 14         14      Paul McCartney 40
## 15         15      George Lucas 233
## 16         16      Elton John 34
## 17         17      David Letterman 40
## 18         18      Phil Mickelson 47
## 19         15      J.K. Rowling 90
## 20         20      Bradd Pitt 25
## 21         21      Peter Jackson 39
## 22         22      Dr. Phil McGraw 45
## 23         23      Jay Leno 32
## 24         24      Celine Dion 40
## 25         25      Kobe Bryant 31
```

```
#d. Access the rows 10 to 20 and save it as Ranks.RData.
```

```
#Write the R script and its output.
```

```
ranked_subset <- celebrity_data[10:20, ]
save(ranked_subset, file = "Ranks.RData")
```

```
#9-a
```

```
library(readxl)
hotels_data <- read_excel("hotels-vienna.xlsx")
hotels_data
```

```
## # A tibble: 428 x 24
##   country city_actual rating_count center1label center2label neighbourhood
##   <chr>    <chr>         <chr>         <chr>         <chr>         <chr>
## 1 Austria Vienna      36           City centre Donauturm    17. Hernals
## 2 Austria Vienna     189           City centre Donauturm    17. Hernals
## 3 Austria Vienna      53           City centre Donauturm    Alsergrund
## 4 Austria Vienna      55           City centre Donauturm    Alsergrund
## 5 Austria Vienna      33           City centre Donauturm    Alsergrund
## 6 Austria Vienna      25           City centre Donauturm    Alsergrund
## 7 Austria Vienna      57           City centre Donauturm    Alsergrund
## 8 Austria Vienna     161           City centre Donauturm    Alsergrund
## 9 Austria Vienna      50           City centre Donauturm    Alsergrund
## 10 Austria Vienna     NA           City centre Donauturm    Alsergrund
## # i 418 more rows
## # i 18 more variables: price <dbl>, city <chr>, stars <dbl>, ratingta <chr>,
## #   ratingta_count <chr>, scarce_room <dbl>, hotel_id <dbl>, offer <dbl>,
## #   offer_cat <chr>, year <dbl>, month <dbl>, weekend <dbl>, holiday <dbl>,
## #   distance <dbl>, distance_alter <dbl>, accommodation_type <chr>,
## #   nnights <dbl>, rating <chr>
```

#b

```
dataset_dimensions <- dim(hotels_data)
dataset_dimensions
```

```
## [1] 428 24
```

#c

```
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
```

```
selected_columns <- hotels_data %>% select(country, neighbourhood, price, stars, accommodation_type, rating)
selected_columns
```

```
## # A tibble: 428 x 6
##   country neighbourhood price stars accommodation_type rating
##   <chr>    <chr>         <dbl> <dbl> <chr>         <chr>
## 1 Austria 17. Hernals      81     4 Apartment    4.4000000000000004
## 2 Austria 17. Hernals      81     4 Hotel        3.9
## 3 Austria Alsergrund      85     4 Hotel        3.7
## 4 Austria Alsergrund      83     3 Hotel        4
## 5 Austria Alsergrund      82     4 Hotel        3.9
## 6 Austria Alsergrund     229     5 Apartment    4.8
```

```
## 7 Austria Alsergrund      103      4 Hotel      3.9
## 8 Austria Alsergrund      150      4 Hotel      4.5999999999999996
## 9 Austria Alsergrund       80      2 Hotel      3.5
## 10 Austria Alsergrund     153      3 Apartment    NA
## # i 418 more rows
```

```
#d
```

```
library(readxl)
library(dplyr)
```

```
hotels_data <- read_excel("hotels-vienna.xlsx")
colnames(hotels_data)
```

```
## [1] "country"      "city_actual"   "rating_count"
## [4] "center1label" "center2label"  "neighbourhood"
## [7] "price"        "city"          "stars"
## [10] "ratingta"     "ratingta_count" "scarce_room"
## [13] "hotel_id"     "offer"         "offer_cat"
## [16] "year"        "month"         "weekend"
## [19] "holiday"     "distance"      "distance_alter"
## [22] "accommodation_type" "nights"       "rating"
```

```
selected_columns <- hotels_data %>% select(country, neighbourhood, price, stars, accommodation_type, rating)
save(selected_columns, file = "new.RData")
```

```
#e
```

```
load("new.RData")
```

```
first_six_rows <- head(selected_columns)
print(first_six_rows)
```

```
## # A tibble: 6 x 6
##   country neighbourhood price stars accommodation_type rating
##   <chr>    <chr>      <dbl> <dbl> <chr>          <chr>
## 1 Austria 17. Hernals      81     4 Apartment  4.4000000000000004
## 2 Austria 17. Hernals      81     4 Hotel      3.9
## 3 Austria Alsergrund      85     4 Hotel      3.7
## 4 Austria Alsergrund      83     3 Hotel      4
## 5 Austria Alsergrund      82     4 Hotel      3.9
## 6 Austria Alsergrund     229     5 Apartment  4.8
```

```
last_six_rows <- tail(selected_columns)
print(last_six_rows)
```

```
## # A tibble: 6 x 6
##   country neighbourhood price stars accommodation_type rating
##   <chr>    <chr>      <dbl> <dbl> <chr>          <chr>
## 1 Austria Wieden         73     3 Hotel      3.4
## 2 Austria Wieden        109     3 Apartment  5
## 3 Austria Wieden        185     5 Hotel      4.3
## 4 Austria Wieden        100     4 Hotel      4.4000000000000004
## 5 Austria Wieden         58     3 Hotel      3.2
## 6 Austria Wieden        110    3.5 Apartment  4
```

```
#10-a
```

```
vegetables <- c("garlic", "corn", "unions", "eggplant", "broccoli", "collard greens", "peas", "cabbage")

#b
vegetables <- c("garlic", "corn", "unions", "eggplant", "broccoli", "collard greens", "peas", "cabbage")

#c
additionalvegetables <-c("mint","patato","cucumber","garlic")
vegetableslist <-append(vegetables,additionalvegetables, after = 5)

#d
vegetableslist <- vegetableslist[-c(15, 10, 5)]
length(vegetableslist)

## [1] 13
```