

RWorksheet__Doronila#2.R

Jocedel Garnette Doronila

2023-12-22

```
vector_a <- -5:5  
vector_a
```

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

```
x <- 1:7  
x
```

```
## [1] 1 2 3 4 5 6 7
```

```
vector_b <- seq(1, 3, by = 0.2)  
vector_b
```

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

```
ages <- 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)  
ages
```

```
## [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
```

```
ages_a <- ages[3]  
ages_a
```

```
## [1] 22
```

```
ages_b <- ages[c(2, 4)]  
ages_b
```

```
## [1] 28 36
```

```
ages_c <- ages[-c(4, 12)]  
ages_c
```

```
## [1] 34 28 22 27 18 52 39 42 29 35 27 22 37 34 19 20 57 49 50 37 46 25 17 37 43  
## [26] 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
```

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

```
## first second third  
##      3      0      9
```

```
names_a <- names[c("first", "third")]  
names_a
```

```
## first third  
##      3      9
```

```
x_2 <- -3:2
x_2[2] <- 0
x_2
```

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

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

```
##   Month Price_Per_Liter_PHP Purchase_Quantity_Liter
## 1   Jan                52.50                    25
## 2   Feb                57.25                    30
## 3 March                60.00                    40
## 4   Apr                65.00                    50
## 5   May                74.25                    10
## 6   June                54.00                    45
```

```
average_fuel <- weighted.mean(diesel_purchase$Price_Per_Liter_PHP, diesel_purchase$Purchase_Quantity_Liter)
average_fuel
```

```
## [1] 59.2625
```

```
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
```

```
length_rivers <- length(rivers)
sum_rivers <- sum(rivers)
mean_rivers <- mean(rivers)
median_rivers <- median(rivers)
var_rivers <- var(rivers)
sd_rivers <- sd(rivers)
min_rivers <- min(rivers)
max_rivers <- max(rivers)
river_data <- c(length_rivers, sum_rivers, mean_rivers, median_rivers, var_rivers, sd_rivers, min_rivers, max_rivers)
river_data
```

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

```
Forbes_Ranking <- data.frame(
  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", "Brad 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, 32, 40, 31
)
)
Forbes_Ranking$Power_Ranking[Forbes_Ranking$Celebrity_Name == "J.K. Rowling"] <- 15
Forbes_Ranking$Pay[Forbes_Ranking$Celebrity_Name == "J.K. Rowling"] <- 90
Forbes_Ranking

```

```

##      Power_Ranking      Celebrity_Name 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      Brad 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

```

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

```
print("CSV file 'PowerRanking.csv' has been saved.")
```

```
## [1] "CSV file 'PowerRanking.csv' has been saved."
```

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

```
print(excel)
```

```

##      Power_Ranking      Celebrity_Name 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     Brad 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
```

```
selected_rows <- Forbes_Ranking[10:20, ]

save(selected_rows, file = "Ranks.RData")

print("RData file 'Ranks.RData' has been saved.")

## [1] "RData file 'Ranks.RData' has been saved."

load("Ranks.RData")

print(selected_rows)
```

```
##      Power_Ranking      Celebrity_Name Pay
## 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      Brad Pitt 25
```

E. The specific output will include the values in columns "Power_Ranking," "Celebrity_Name," and "Pay"

```
library(readr)
hotels_vienna <- read_csv("hotels-vienna.xlsx.csv")
```

```
## Rows: 428 Columns: 24
## -- Column specification -----
```

```
## Delimiter: ","
## chr (8): country, city_actual, center1label, center2label, neighbourhood, c...
## dbl (16): rating_count, price, stars, ratingta, ratingta_count, scarce_room,...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
dim(hotels_vienna)

## [1] 428 24
selected_columns <- hotels_vienna[, c("country", "neighbourhood", "price", "stars", "accommodation_type", "rating")]
save(selected_columns, file = "new.RData")
load("new.RData")
head(selected_columns)

## # A tibble: 6 x 6
##   country neighbourhood price stars accommodation_type rating
##   <chr>    <chr>         <dbl> <dbl> <chr>          <dbl>
## 1 Austria 17. Hernals      81    4 Apartment      4.4
## 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
tail(selected_columns)

## # A tibble: 6 x 6
##   country neighbourhood price stars accommodation_type rating
##   <chr>    <chr>         <dbl> <dbl> <chr>          <dbl>
## 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          4.4
## 5 Austria Wieden          58    3 Hotel          3.2
## 6 Austria Wieden         110   3.5 Apartment      4
vegetables <- c("Carrot", "Broccoli", "Spinach", "Tomato", "Cucumber", "Bell Pepper", "Zucchini", "Lettuce")
print(vegetables)

## [1] "Carrot"      "Broccoli"    "Spinach"     "Tomato"      "Cucumber"
## [6] "Bell Pepper" "Zucchini"    "Lettuce"     "Cabbage"     "Onion"
vegetables <- c(vegetables, "Kale", "Eggplant")
print(vegetables)

## [1] "Carrot"      "Broccoli"    "Spinach"     "Tomato"      "Cucumber"
## [6] "Bell Pepper" "Zucchini"    "Lettuce"     "Cabbage"     "Onion"
## [11] "Kale"        "Eggplant"
vegetables <- append(vegetables, c("Radish", "Artichoke", "Asparagus", "Cauliflower"), after = 5)
print(vegetables)

## [1] "Carrot"      "Broccoli"    "Spinach"     "Tomato"      "Cucumber"
```

```
## [6] "Radish"      "Artichoke"    "Asparagus"   "Cauliflower" "Bell Pepper"
## [11] "Zucchini"    "Lettuce"      "Cabbage"      "Onion"        "Kale"
## [16] "Eggplant"
```

```
length(vegetables)
```

```
## [1] 16
```

```
vegetables <- vegetables[-c(5, 10, 15)]
```

```
print(vegetables)
```

```
## [1] "Carrot"      "Broccoli"     "Spinach"      "Tomato"       "Radish"
## [6] "Artichoke"   "Asparagus"   "Cauliflower"  "Zucchini"     "Lettuce"
## [11] "Cabbage"     "Onion"        "Eggplant"
```

```
length(vegetables)
```

```
## [1] 13
```