

# RWorksheets\_Madayag#2

MJ MADAYAG

2024-09-25

## NO. 1

```
## A.  
## it displayed numbers from -5 to positive 5  
vector_1 <- -5:5  
vector_1
```

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

```
## B.  
x <- 1:7  
x
```

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

## NO. 2

```
vector_2 <- seq(1, 3, by = 0.2)  
vector_2
```

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

```
##incremented numbers by .2 from 1 till 3
```

## NO. 3

```
## A.  
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)
```

```
third_element <- ages[3]  
third_element
```

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

```
## B.  
second_and_fourth <- ages[c(2, 4)]  
second_and_fourth
```

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

```
## C.
all_but_4th_and_12th <- ages[-c(4, 12)]
all_but_4th_and_12th

## [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
```

## NO. 4

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

selected_elements <- x[c("first", "third")]

selected_elements
```

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

*##shows or accessed the first and third element*

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

selected_elements <- x[c("first", "third")]

selected_elements
```

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

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

## NO. 5

```
## A.
x <- -3:2

x[2] <- 0

x
```

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

*## the second element in this sequence is changed to 0*

```
##B.
x <- -3:2

x[2] <- 0

x
```

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

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

no 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),
  Liters = c(25, 30, 40, 50, 10, 45)
)
diesel_fuel
```

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

```
## B.
weighted.mean(diesel_fuel$liter, diesel_fuel$purchase)
```

```
## [1] NaN
```

NO. 7

```
## A. and B. and C.
```

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									

```
data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers), var(rivers), sd(river
```

```
data
```

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

## NO. 8

## A.

```
power_rank <- c(1:25)
```

```
celeb_name <- c("Tom Cruise", "Rolling Stones", "Oprah Winfrey", "U2", "Tiger Woods", "Ste
```

```
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)
```

```
celeb_data <- data.frame(PowerRanking = power_rank, CelebrityName = celeb_name, Pay = pay)
celeb_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 McCatney  40
## 15            15    George Lucas 233
## 16            16    Elthon 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 Lenon  32
## 24            24    Celine Dion  40
## 25            25    Kobe Bryant  31
```

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

```
celeb_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 McCatney  40
## 15            15    George Lucas 233
## 16            16    Elthon 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 Lenon  32
## 24          24          Celine Dion  40
## 25          25          Kobe Bryant 31
```

## C.

*## Create an excel file from the table above and save it  
#as csv file(PowerRanking). Import the csv file into  
#the RStudio. What is the R script?*

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

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

```
celeb_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 McCatney  40
## 15         15      George Lucas 233
## 16         16      Elthon 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 Lenon  32
## 24         24      Celine Dion  40
## 25         25      Kobe Bryant 31
```

## D.

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

## NO. 9

## NO. 9

```
library(readxl)
```

```

# A.
# Import the Excel file
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>

# Get the dimensions of the dataset
# B.
dimensions <- dim(hotels_data)
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

# Select specific columns
selected_data <- hotels_data %>%
  select(country, neighbourhood, price, stars, accommodation_type, rating)

head(selected_data)

## # 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
```

## 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" "nnights"          "rating"
```

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

## E.

```
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
```

NO 10.

## A

```
vegetables <- c("Carrot", "Broccoli", "Spinach", "Bell Pepper",
               "Cauliflower", "Onion", "Tomato", "Eggplant",
```

```

      "Green Bean", "Potato")
vegetables

## [1] "Carrot"      "Broccoli"    "Spinach"     "Bell Pepper" "Cauliflower"
## [6] "Onion"       "Tomato"      "Eggplant"    "Green Bean"  "Potato"

## B
vegetables <- c(vegetables, "Water Spinach", "Squash")
vegetables

## [1] "Carrot"      "Broccoli"    "Spinach"     "Bell Pepper"
## [5] "Cauliflower" "Onion"       "Tomato"      "Eggplant"
## [9] "Green Bean"  "Potato"      "Water Spinach" "Squash"

## C.
vegetables <- append(vegetables, c("Celery", "Pumpkin", "Moringa Oliefera", "Pea"), after = 5)
vegetables

## [1] "Carrot"      "Broccoli"    "Spinach"     "Bell Pepper"
## [5] "Cauliflower" "Celery"      "Pumpkin"     "Moringa Oliefera"
## [9] "Pea"         "Onion"       "Tomato"      "Eggplant"
## [13] "Green Bean"  "Potato"      "Water Spinach" "Squash"

num_vegetables <- length(vegetables)
num_vegetables

## [1] 16

## D.
vegetables <- vegetables[-c(5, 10, 15)]
vegetables

## [1] "Carrot"      "Broccoli"    "Spinach"     "Bell Pepper"
## [5] "Celery"      "Pumpkin"     "Moringa Oliefera" "Pea"
## [9] "Tomato"      "Eggplant"    "Green Bean"  "Potato"
## [13] "Squash"

remaining_vegetables <- length(vegetables)
remaining_vegetables

## [1] 13

```