RWorksheet_Arenal#2

Jo Carlo Arenal

2024-09-25

```r
# 1
# a
sequence_vector <- -5:5

sequence_vector
```

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

```r
# The output is a vector containing all integer values from -5 to 5, inclusive. The : operator generate

# b

x <- 1:7

x
```

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

```r
# 2
# a

sequence_vector_seq <- seq(1, 3, by = 0.2)

sequence_vector_seq
```

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

```r
# The output is a vector that starts at 1 and ends at 3, with increments of 0.2. The seq() function gen

# 3
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
```

```r
# b

second_fourth_elements <- ages[c(2, 4)]

second_fourth_elements
```

```
## [1] 28 36
# c.

ages_excluding_4th_12th <- ages[-c(4, 12)]

ages_excluding_4th_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
# 4
# a

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

print(x)

##  first second  third
##      3      0      9
selected_elements <- x[c("first", "third")]

selected_elements

## first third
##     3     9
# The output shows a vector that contains only the elements corresponding to the names "first" and "thi

# 5
# a
x[2] <- 0
x

##  first second  third
##      3      0      9
# b

x <- -3:2

print(x)

## [1] -3 -2 -1  0  1  2
x[2] <- 0

print(x)

## [1] -3  0 -1  0  1  2
# 6
# a

months <- c("Jan", "Feb", "March", "Apr", "May", "June")
price_per_liter <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)
purchase_quantity <- c(25, 30, 40, 50, 10, 45)
```

```r
diesel_data <- data.frame(Month = months,
                          Price_per_liter = price_per_liter,
                          Purchase_quantity = purchase_quantity)

print(diesel_data)
```

```
##   Month Price_per_liter Purchase_quantity
## 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
```

```r
# b

expenditure <- price_per_liter * purchase_quantity

average_expenditure <- weighted.mean(expenditure, purchase_quantity)

average_expenditure
```

```
## [1] 2298.062
```

```r
# 7
# a
# b
# c

data <- c(length(rivers),      # Number of elements
          sum(rivers),          # Sum of lengths
          mean(rivers),         # Mean of lengths
          median(rivers),       # Median of lengths
          var(rivers),          # Variance
          sd(rivers),           # Standard deviation
          min(rivers),          # Minimum length
          max(rivers))          # Maximum length

data
```

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

```r
# 8
# a

Power_ranking <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24

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,
```

```r
          75, 25, 39, 45, 32, 40, 31)

celebrity_df <- data.frame(Power_ranking = Power_ranking,
                           Celebrity_name = Celebrity_name,
                           Pay = Pay)

celebrity_df
```

```
##    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            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
```

```r
# b
Power_ranking <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 2

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, 32, 40, 31)

celebrity_df <- data.frame(Power_ranking, Celebrity_name, Pay)

celebrity_df[celebrity_df$Celebrity_name == "J.K. Rowling", "Power_ranking"] <- 15
celebrity_df[celebrity_df$Celebrity_name == "J.K. Rowling", "Pay"] <- 90

celebrity_df
```

```
##    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             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
```

```r
# 9
# a
library(readxl)
hotels_vienna <- read_excel("/cloud/project/hotels-vienna.xlsx")
hotels_vienna
```

```
## # 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>
```

```r
# b
dim(hotels_vienna)
```

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

```r
# c
library(magrittr)
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
```

```r
str(hotels_vienna)
```

```
## tibble [428 x 24] (S3: tbl_df/tbl/data.frame)
##  $ country          : chr [1:428] "Austria" "Austria" "Austria" "Austria" ...
##  $ city_actual      : chr [1:428] "Vienna" "Vienna" "Vienna" "Vienna" ...
##  $ rating_count     : chr [1:428] "36" "189" "53" "55" ...
##  $ center1label     : chr [1:428] "City centre" "City centre" "City centre" "City centre" ...
##  $ center2label     : chr [1:428] "Donauturm" "Donauturm" "Donauturm" "Donauturm" ...
##  $ neighbourhood    : chr [1:428] "17. Hernals" "17. Hernals" "Alsergrund" "Alsergrund" ...
##  $ price            : num [1:428] 81 81 85 83 82 229 103 150 80 153 ...
##  $ city             : chr [1:428] "Vienna" "Vienna" "Vienna" "Vienna" ...
##  $ stars            : num [1:428] 4 4 4 3 4 5 4 4 2 3 ...
##  $ ratingta         : chr [1:428] "4.5" "3.5" "3.5" "4" ...
##  $ ratingta_count   : chr [1:428] "216" "708" "629" "52" ...
##  $ scarce_room      : num [1:428] 1 0 0 0 1 1 1 0 1 1 ...
##  $ hotel_id         : num [1:428] 21894 21897 21901 21902 21903 ...
##  $ offer            : num [1:428] 1 1 1 1 1 1 0 0 1 1 ...
##  $ offer_cat        : chr [1:428] "15-50% offer" "1-15% offer" "15-50% offer" "15-50% offer" ...
##  $ year             : num [1:428] 2017 2017 2017 2017 2017 ...
##  $ month            : num [1:428] 11 11 11 11 11 11 11 11 11 11 ...
##  $ weekend          : num [1:428] 0 0 0 0 0 0 0 0 0 0 ...
##  $ holiday          : num [1:428] 0 0 0 0 0 0 0 0 0 0 ...
##  $ distance         : num [1:428] 2.7 1.7 1.4 1.7 1.2 0.9 0.9 1 0.7 1.5 ...
##  $ distance_alter   : num [1:428] 4.4 3.8 2.5 2.5 2.8 3 2.4 2.7 2.7 2.7 ...
##  $ accommodation_type: chr [1:428] "Apartment" "Hotel" "Hotel" "Hotel" ...
##  $ nnights          : num [1:428] 1 1 1 1 1 1 1 1 1 1 ...
##  $ rating           : chr [1:428] "4.4000000000000004" "3.9" "3.7" "4" ...
```

```r
# d


# e



# 10
# a
vegetables <- c("Carrot", "Broccoli", "Spinach", "Potato", "Tomato",
                "Cabbage", "Bell Pepper", "Zucchini", "Onion", "Eggplant")


vegetables
```

```
##  [1] "Carrot"      "Broccoli"    "Spinach"    "Potato"      "Tomato"
##  [6] "Cabbage"     "Bell Pepper" "Zucchini"   "Onion"       "Eggplant"
```

```r
# b
vegetables <- c(vegetables, "Cauliflower", "Radish")

vegetables
```

```
##  [1] "Carrot"      "Broccoli"    "Spinach"    "Potato"      "Tomato"
##  [6] "Cabbage"     "Bell Pepper" "Zucchini"   "Onion"       "Eggplant"
## [11] "Cauliflower" "Radish"
```

```r
# c
vegetables <- append(vegetables, c("Asparagus", "Kale", "Lettuce", "Mushroom"), after = 5)

vegetables
```

```
##  [1] "Carrot"      "Broccoli"    "Spinach"    "Potato"      "Tomato"
##  [6] "Asparagus"   "Kale"        "Lettuce"    "Mushroom"    "Cabbage"
## [11] "Bell Pepper" "Zucchini"    "Onion"      "Eggplant"    "Cauliflower"
## [16] "Radish"
```

```r
length(vegetables)
```

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

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

vegetables
```

```
##  [1] "Carrot"      "Broccoli"    "Spinach"    "Potato"      "Asparagus"
##  [6] "Kale"        "Lettuce"     "Mushroom"   "Bell Pepper" "Zucchini"
## [11] "Onion"       "Eggplant"    "Radish"
```

```r
length(vegetables)
```

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