RWorksheet_Calzado#1.Rmd.

Michael Angelo S. Calzado

2024-09-04

Set up a vector named age, consisting of 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, 42, 53, 41, 51, 35, 24, 33, 41.

a. How many data points?
34

b. Write the R code and its output.

```
age <- 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, 42, 53, 41, 51,
         35, 24, 33, 41)
length(age)
```

```
## [1] 34
```

2. Find the reciprocal of the values for age.

Write the R code and its output.

```
library(MASS)
fractions(reciprocal_age <- 1 / age)
```

```
##  [1] 1/34 1/28 1/22 1/36 1/27 1/18 1/52 1/39 1/42 1/29 1/35 1/31 1/27 1/22 1/37
## [16] 1/34 1/19 1/20 1/57 1/49 1/50 1/37 1/46 1/25 1/17 1/37 1/42 1/53 1/41 1/51
## [31] 1/35 1/24 1/33 1/41
```

3. Assign also new_age <- c(age, 0, age).

What happen to the new_age?

```
new_age <- c(age, 0, age)

print(new_age)
```

```
##  [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 42 53 41 51 35 24 33 41  0 34 28 22 36 27 18 52 39 42 29 35 31 27 22 37
## [51] 34 19 20 57 49 50 37 46 25 17 37 42 53 41 51 35 24 33 41
```

4. Sort the values for age.

Write the R code and its output.

```
sort(age)
```

```
##  [1] 17 18 19 20 22 22 24 25 27 27 28 29 31 33 34 34 35 35 36 37 37 37 39 41 41
## [26] 42 42 46 49 50 51 52 53 57
```

5. Find the minimum and maximum value for age.

Write the R code and its output.

```
min_age <- min(age)
max_age <- max(age)
print(paste("Minimum value:", min_age))
```

```
## [1] "Minimum value: 17"
```

```
print(paste("Maximum value:", max_age))
```

```
## [1] "Maximum value: 57"
```

6. Set up a vector named data, consisting of 2.4, 2.8, 2.1, 2.5, 2.4, 2.2, 2.5, 2.3, 2.5, 2.3, 2.4, and 2.7.

a. How many data points?
b. Write the R code and its output.

```
data <- c(2.4, 2.8, 2.1, 2.5, 2.4, 2.2, 2.5, 2.3, 2.5, 2.3, 2.4, 2.7)

number_of_data_points <- length(data)

print(paste("Number of data points:", number_of_data_points))
```

```
## [1] "Number of data points: 12"
```

7. Generates a new vector for data where you double every value of the data. What happen to the data?

```
doubled_data <- data * 2

print("Original data:")
```

```
## [1] "Original data:"
```

```
print(data)
```

```
##  [1] 2.4 2.8 2.1 2.5 2.4 2.2 2.5 2.3 2.5 2.3 2.4 2.7
```

```
print("Doubled data:")
```

```
## [1] "Doubled data:"
```

```
print(doubled_data)
```

```
##  [1] 4.8 5.6 4.2 5.0 4.8 4.4 5.0 4.6 5.0 4.6 4.8 5.4
```

8. Generate a sequence for the following scenario:
9. Integers from 1 to 100.
10. Numbers from 20 to 60
11. Mean of numbers from 20 to 60
12. Sum of numbers from 51 to 91
13. Integers from 1 to 1,000

a. How many data points from 8.1 to 8.4?_____
b. Write the R code and its output from 8.1 to 8.4.

```
sequence_1_to_100 <- 1:100
sequence_20_to_60 <- 20:60
mean_20_to_60 <- mean(sequence_20_to_60)
sequence_51_to_91 <- 51:91
```

```r
sum_51_to_91 <- sum(sequence_51_to_91)
sequence_1_to_1000 <- 1:1000
num_data_points_1_to_100 <- length(sequence_1_to_100)
num_data_points_20_to_60 <- length(sequence_20_to_60)
num_data_points_51_to_91 <- length(sequence_51_to_91)
num_data_points_1_to_1000 <- length(sequence_1_to_1000)

num_data_points_8_1_to_8_4 <- num_data_points_1_to_100 + num_data_points_20_to_60 + num_data_points_51_

print(paste("Number of data points from 8.1 to 8.4:", num_data_points_8_1_to_8_4))
```

```
## [1] "Number of data points from 8.1 to 8.4: 182"
```

```r
print(paste("Number of data points from 1 to 100:", num_data_points_1_to_100))
```

```
## [1] "Number of data points from 1 to 100: 100"
```

```r
print(paste("Number of data points from 20 to 60:", num_data_points_20_to_60))
```

```
## [1] "Number of data points from 20 to 60: 41"
```

```r
print(paste("Number of data points from 51 to 91:", num_data_points_51_to_91))
```

```
## [1] "Number of data points from 51 to 91: 41"
```

```r
print(paste("Number of data points from 1 to 1,000:", num_data_points_1_to_1000))
```

```
## [1] "Number of data points from 1 to 1,000: 1000"
```

```r
print(paste("Mean of numbers from 20 to 60:", mean_20_to_60))
```

```
## [1] "Mean of numbers from 20 to 60: 40"
```

```r
print(paste("Sum of numbers from 51 to 91:", sum_51_to_91))
```

```
## [1] "Sum of numbers from 51 to 91: 2911"
```