RWorksheet_calvario#1.Rmd

Jolien

2024-11-06

1. Set up vector named age, consisting 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.

```r
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, 1
length(age)
```

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

How many data points?

34 data points

2. Find the reciprocal of the values for age. Write the R code and its output.

```r
reciprocal <- 1/age
print(reciprocal)
```

```
##  [1] 0.02941176 0.03571429 0.04545455 0.02777778 0.03703704 0.05555556
##  [7] 0.01923077 0.02564103 0.02380952 0.03448276 0.02857143 0.03225806
## [13] 0.03703704 0.04545455 0.02702703 0.02941176 0.05263158 0.05000000
## [19] 0.01754386 0.02040816 0.02000000 0.02702703 0.02173913 0.04000000
## [25] 0.05882353 0.02702703 0.02380952 0.01886792 0.02439024 0.01960784
## [31] 0.02857143 0.04166667 0.03030303 0.02439024
```

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

```r
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
```

What happen to the new_age?

The resulting vector new_age will contain the values of age, followed by 0, and then age again.

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)
```

```
## [1] 17
```

```
max (age)
```

```
## [1] 57
```

17 is the minimum

57 is the maximum

6.

{r} 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)

{r} length(data) # 12 data points

7.

the vector results in a new vector where each element is twice the original element.

{r} new_vector <-(data) data * 2 # 8. Generate a sequence for the following scenario: # 8.1 - Integers from 1 to 100. {r} a <- seq(1:100) 8.2 - Numbers from 20 to 60. {r} l <- seq(20,60) 8.3 - Mean of numbers from 20 to 60. {r} i <- mean(20,60) 8.4 - Sum of numbers from 51 to 91. {r} k <- sum(51:91) 8.5 - Integers from 1 to 1,000. {r} t <- seq(1:1000)

a. How many data points from 8.1 to 8.4?

answer: 143 data points

{r} length1 <- length(a) length2 <- length(l) length3 <- length(i) length4 <- length(k) sum (length1+length2+length3+length4) # b. Write the R code and its output from 8.1 to 8.4. {r} a

<- seq(1:100) l <- seq(20,60) i <- mean(20,60) k <- sum(51:91) # c. For 8.5 find only maximum data points until 10. {r} t <- 1:1000 answer <- max(t[t <- 10]) answer

9. *Print a vector with the integers between 1 and 100 that are not divisible by 3, 5 and 7 using filter option.

{r} Filter(function(i) { all(i %% c(3,5,7) != 0) }, seq(100))

10. Generate a sequence backwards of the integers from 1 to 100.

{r} rev(seq(1:100))

11. List all the natural numbers below 25 that are multiples of 3 or 5.

{r} upper_limit <- 25 num <- 1:(upper_limit - 1) multiples <- num[num %% 3 == 0 | num %% 5 == 0] multiples # a. How many data points from 10 to 11? # answer: 11 data points {r} length(multiples) # b. Write the R code and its output from 10 and 11. {r} dp10 <- length(multiples) dp11 <- 1

total <- dp10 + dp11 total

12.

{r} x <- {0 + x + 5 + 3}

Describe the output.

answer: he line is incomplete. Thus, it caused an error.

13. Set up a vector named score

{r} score <- c(72, 86, 92, 63, 88, 89, 91, 92, 75, 75, 77)

x2 <- score[2] x3 <- score[3]

print(x2) print(x3)

14. Create a vector

{r} a <- c(1,2,NA,4,NA,6,7)

print(a, na.print = "-999") # b. describe the output. # answer: The output displays the vector a with NA values shown as -999, while the actual vector remains unchanged with NA values.

15.

{r} name = readline(prompt="Input your name:") age = readline(prompt="Input your age:") print(paste("My name is",name, "and I am",age ,"years old.")) print(R.version.string)