WorkSheet - 1

John Laurence Andica

2024-09-04

1.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.

```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
```

   a. How many data points? 34

```r
length(age)
```

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

   b. Write the R code and its output.

```r
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
```

   2. Find the reciprocal of the values for age.

```r
recip <- 1 / age
```

Write the R code and its output.

```r
fractions(recip)
```

```
##  [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).

```r
new_age <- c(age, 0, age)
#It added a 0 in the middle, andIt assigns the value of age to new_age
```

   4. Sort the values for age.

```r
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 maximum and minimum value for age"

```r
max(age)
```

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

```r
min(age)
```

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

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.

```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)
```

    a. How many data points?

```r
length(data)
```

```
## [1] 12
```

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

```r
new_data <- data * 2
#The value of the elements in data was doubled
```

8. Generate the following scenario

```r
s1 <-seq(1:100)
#8.2
s2 <- seq(20:60)
#8.3
m <- mean(s2)
#8.4
add1 <- sum(51:91)
#8.5
s3 <- seq(1:1000)
#a.
allLength <- length(s1) + length(s2) + length(m) + length(add1)
allLength
```

```
## [1] 143
```

```r
#c
max(s3[s3 <- 10])
```

```
## [1] 10
```

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))
```

```
##  [1]  1  2  4  8 11 13 16 17 19 22 23 26 29 31 32 34 37 38 41 43 44 46 47 52 53
## [26] 58 59 61 62 64 67 68 71 73 74 76 79 82 83 86 88 89 92 94 97
```

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

```r
s10 <- seq(100, 1)
s10
```

```
##   [1] 100  99  98  97  96  95  94  93  92  91  90  89  88  87  86  85  84  83
##  [19]  82  81  80  79  78  77  76  75  74  73  72  71  70  69  68  67  66  65
##  [37]  64  63  62  61  60  59  58  57  56  55  54  53  52  51  50  49  48  47
##  [55]  46  45  44  43  42  41  40  39  38  37  36  35  34  33  32  31  30  29
##  [73]  28  27  26  25  24  23  22  21  20  19  18  17  16  15  14  13  12  11
##  [91]  10   9   8   7   6   5   4   3   2   1
```

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

```r
vec <- c(24:1)
s <- Filter(function(i) { all(i %% c(3) == 0 )| all(i %% c(5) == 0)}, vec)
sum(s)
```

```
## [1] 143
```

```
#a. How many data points from 10 to 11? 111"
length(s10) + length(s)
```

```
## [1] 111
```

12. Describe the output

```
#x <- {0 + x + 5 + }
#The error occured because of the + operator,it expecting something to be added to it but nothing is th
```

13. *Set up a vector named score, consisting of 72, 86, 92, 63, 88, 89, 91, 92, 75, 75 and 77. To access individual elements of an atomic vector, one generally uses the x[i] construction.

```
score <- c(72, 86, 92, 63, 88, 89, 91, 92, 75 ,75, 77)
#Find x[2]
score[2]
```

```
## [1] 86
```

```
#Find x[3]
score[3]
```

```
## [1] 92
```

14. *Create a vector a = c(1,2,NA,4,NA,6,7).

```
a = c(1,2,NA,4,NA,6,7)
```

a. Change the NA to 999 using the codes print(a,na.print="-999").

```
print(a,na.print="-999")
```

```
## [1]    1    2 -999    4 -999    6    7
```

b. Write the R code and its output. Describe the output. The elements in vector that has a value of NA was replaced by the value of -999 because of the code above.

15. A special type of function calls can appear on the left hand side of the assignment operator as in > class(x) <- "foo".

```
name = readline(prompt="Input your name: ")
```

```
## Input your name:
age = readline(prompt="Input your age: ")
```

```
## Input your age:
print(paste("My name is",name, "and I am",age ,"years old."))
```

```
## [1] "My name is  and I am  years old."
print(R.version.string)
```

```
## [1] "R version 4.4.1 (2024-06-14)"
```

the output of the code above is "Input your name" when you enter your name it asks "Input your age" then when you enter your age it will print the "R version 4.4.1 (2023-06-14) in the next line. The code prompts the user their name and age, it also display the R version.