```
#CS101 Worksheet-1 in R
#Sicabalo, Mark Lexter BSIT 2-A

#1.
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)

#a.
length(age)
#answer: 34 data points

#b.
age

#2.Reciprocal value for age.
reciprocal <- function(age) vec <- 1/age
rage <- reciprocal(age)
rage

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

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

#answer: it will display number with 0 in the center.

#4. Sort the values for age.
sort(age)
#write the code and it's output.
#17 18 19 20 22 22 24 25 27 27 28 29 31 33 34 34 35 35 36 37 37
#[22] 37 39 41 41 42 42 46 49 50 51 52 53 57

#5. find the min and max for age.
max(age) #max is 57
min(age) #min is 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.
vec_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)

#how many data points?
length(vec_data)
#answer: 12

#b Write its R code and its output
vec_data
#Output: 2.4 2.8 2.1 2.5 2.4 2.2 2.5 2.3 2.5 2.3 2.4 2.7

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

2*vec_data
#answer: the data will double
#output: 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:
#8.1
c(1:100)
#answer/output: [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 [19] 19 20 21 22 23 24 25 26
27 28 29 30 31 32 33 34 35 36 [37] 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
[55] 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 [73] 73 74 75 76 77 78 79 80 81
82 83 84 85 86 87 88 89 90 [91] 91 92 93 94 95 96 97 98 99 100
#8.2
num <- c(20:60)
num
#answer/output: [1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
43 44 [26] 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
#*8.3 Mean of numbers from 20 to 60
mean(num)
#answer: mean is 40
#*8.4 Sum of numbers from 51 to 91
num_sum <- c(51:91)
sum(num_sum)
#answer: sum is 2911
#*8.5 Integers from 1 to 1,000
c(1:1000)
#a. answer: 43 data points are in 8.1 to 8.4
#b.Write the R code and its output from 8.1 to 8.4.
seq(1:100)
#Output is number sequence from 1-100.

x <- c(20:60)
print(seq(x))#Output is numbers 1 -41.
print(mean(x))#output is 40
print(sum(51:91))# output is 2911
#c. For 8.5 find only maximum data points until 10.
m <- seq(1:10)
max(m)
#answer is 10
#9. *Print a vector with the integers between 1 and 100 that are not divisible by 3, 5 and 7
#using filter option
Filter(function(i) { all(i %% c(3,5,7) != 0) }, seq(100))

#10. Generate a sequence backwards of the integers from 1 to 100.
#Write the R code and its output.
seq(from = 100, to = 1)

# 11.	List all the natural numbers below 25 that are multiples of 3 or 5.
sum((1:25)[((1:25)%%3 == 0) | ((1:25)%%5 == 0)])

#a. How many data points from 10 to 11?
101
#b. Write the R code and its output from 10 and 11.
seq(from = 100, to = 1) #output is numbers from 100 to 1
sum((1:25)[((1:25)%%3 == 0) | ((1:25)%%5 == 0)])#output is 168
#12 Enter this statement:
# { x <- 0+ x + 5 + }
#Answer : Error

#13
score <- c(72, 86, 92, 63, 88, 89, 91, 92, 75,75, 77)
# Answer: x[2] = 86 x[3] = 92
```

```
#14
a <- c(1,2,NA,4,NA,6,7)
print(a,na.print="-999")

#15
class(x) <- "foo"
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)
```