

“WORKSHEET-2 in R”

DRAKE FRANCIS M. JACULINA

2022-10-07

BSIT 2-A

Worksheet for R Programming

```
#1 Create a vector using : operator  
#a. Sequence from -5 to 5. Write the R code and its output. Describe its output.  
Seq_operator <- -5:5  
print(Seq_operator)
```

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

```
#b. x <- 1:7. What will be the value of x?  
x <- 1:7  
print(x)
```

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

```
#2 Create a vector using seq() function  
#a. seq(1, 3, by=0.2) # specify step size  
#Write the R code and its output. Describe the output.  
y = seq(1, 3, by=0.2)  
print(y)
```

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

```
#It generated a 0.2 sequence from 1 to 3. It increases by 0.2 till it reaches 3.
```

```
#3 A factory has a census of its workers. There are 50 workers in total.  
#The following list shows their ages:  
#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.  
#a. Access 3rd element, what is the value?  
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, 43, 53, 41, 51, 35, 24, 33,  
         41, 53, 40, 18, 44, 38, 41, 48, 27, 39, 19,  
         30, 61, 54, 58, 26, 18)  
print(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 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
```

```
print(age)[3]
```

```
## [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 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
## [1] 22
```

```
#b. Access 2nd and 4th element, what are the values?
print(age)[2]
```

```
## [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 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
## [1] 28
```

```
print(age)[4]
```

```
## [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 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
## [1] 36
```

```
#c. Access all but the 1st element is not included. Write the R code and its output.
print(age)[2:50]
```

```
## [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 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
## [1] 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
## [26] 43 53 41 51 35 24 33 41 53 40 18 44 38 41 48 27 39 19 30 61 54 58 26 18
```

```
#4 Create a vector x <- c("first"=3, "second"=0, "third"=9). Then named the vector, names(x).
x <- c("first"=3, "second"=0, "third"=9)
names(x)
```

```
## [1] "first" "second" "third"
```

```
#a. Print the results. Then access x[c("first","third")] Describe the output.
print(names(x))
```

```
## [1] "first" "second" "third"
```

```
x[c("first","third")]
```

```
## first third
##      3      9
```

```
#b. Write the code and its output.
x[c("first","third")]
```

```
## first third
##      3      9
```

```
#After printing, a table with the first, second, and third values
#redistributed to each character is provided. The same table is next,
#except for the second one is no longer exists.
```

```
#5 Create a sequence x from -3:2.
x <- seq(-3:2)
print(x)
```

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

```
#a. Modify 2nd element and change it to 0;
#Describe the output.
#b. Write the code and its output.
```

```
x[2] <- 0
print(x)
```

```
## [1] 1 0 3 4 5 6
```

*#The sequence was 1 to 6 after changing x [2] into 0, whereas it is now
#1 0 3 4 5 6 after changing x [2] into 0. Now, 0 is in place of 2.*

*#6. *The following data shows the diesel fuel purchased by Mr. Cruz.*

```
#a. Create a data frame for month, price per liter (php)
#and purchase-quantity (liter). Write the codes.
Months <- c("Jan", "Feb", "March", "April", "May", "June")
Price_per_liter_php <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)
Purchase_quantity_liter <- c(25, 30, 40, 50, 10, 45)
data_frame <- data.frame(Months, Price_per_liter_php, Purchase_quantity_liter)
print (data_frame)
```

```
##   Months Price_per_liter_php Purchase_quantity_liter
## 1   Jan                52.50                    25
## 2   Feb                57.25                    30
## 3 March                60.00                    40
## 4 April                65.00                    50
## 5   May                74.25                    10
## 6   June               54.00                    45
```

```
#b. What is the average fuel expenditure of Mr. Cruz
#from Jan to June? Note: Use weighted.mean(liter, purchase)
average_fuel <- weighted.mean(Price_per_liter_php, Purchase_quantity_liter)
print (average_fuel)
```

```
## [1] 59.2625
```

*#7. R has actually lots of built-in data sets. For example, the rivers data "gives the lengths
(in miles) of 141 "major" rivers in North America, as compiled by the US Geological Survey".*

```
#a. Type "rivers" in your R console. Create a vector data with 7
#elements, containing the number of elements (length) in rivers,
#their sum (sum), mean (mean), median (median), variance (var)
#standard deviation (sd), minimum (min) and maximum (max).
data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers), var(rivers),
          sd(rivers), min(rivers), max(rivers))
data
```

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

```
#b. What are the results?
#The results are length, sum, mean, median, var, sd, min, max of the rivers data.
#c. Write the code and its outputs.
```

*#8. The table below gives the 25 most powerful celebrities and their annual pay as ranked
#by the editions of Forbes magazine and as listed on the Forbes.com website.*

```
#a. Create vectors according to the above table. Write the codes.
Ranking <- 1:25
Celebrities <- 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 Lenon", "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)
Forbes_Rangking <- data.frame(Ranking, Celebrities, Pay)
print (Forbes_Rangking)

```

```

##      Ranking      Celebrities 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 Lenon  32
## 24         24    Celine Dion  40
## 25         25    Kobe Bryant  31

```

#b. Modify the power ranking and pay of J.K. Rowling. Change power ranking to 15 and #pay to 90. Write the codes and its output.

```

Ranking [19] <- 15
print (Ranking)

```

```

## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 15 20 21 22 23 24 25

```

```

Pay [19] <- 90
print (Pay)

```

```

## [1] 67 90 225 110 90 332 302 41 52 88 55 44 55 40 233 34 40 47 90
## [20] 25 39 45 32 40 31

```

```

Forbes_Ranking <- data.frame(Ranking, Celebrities, Pay)
Forbes_Ranking

```

```

##      Ranking      Celebrities 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 Lenon	32
## 24	24	Celine Dion	40
## 25	25	Kobe Bryant	31

#c. Interpret the data.

#The position and pay of J.K. Rowling was changed.

#From position 19 it change the power ranking into 15 and the 75 pay it change into 90.