Worksheet 2

Grace Anne E. Capanang

2022-10-07

Instructions:

• Use RStudio or the RStudio Cloud accomplish this worksheet. + Save the R script as RWorksheet_lastname#2.R. • Create your own GitHub repository and push the R script as well as this pdf worksheet to your own repo.

Accomplish this worksheet by answering the questions being asked and writing the code manually.

Using Vectors

1. Create a vector using: operator

a. Sequence from -5 to 5.

```
five <- -5:5
five
```

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

The output produces values from numbers -5 to 5.

b. x <- 1:7. What will be the value of x?

```
x <- 1:7
x
```

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

The value of x are numbers from 1 to 7 in sequence, that is, 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.

```
jump <- seq(1, 3, by = 0.2)
jump
```

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

The output is numbers from 1 to 3 sequently with a decimal 0.2 in between, that is 1.0 1.2 1.4 1.6 1.8 2.0 2.2 2.4 2.6 2.8 3.0.

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.

```
    workersAges <- 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)
    workersAges
```

```
##  [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
```

3a. Access 3rd element, what is the value?

```
    ThirdElement <- workersAges[3]
    ThirdElement
```

```
## [1] 22
```

The value in the 3rd element is 22.

3b. Access 2nd and 4th element, what are the values?

```
    SecondElement <- workersAges[2]
    SecondElement
```

```
## [1] 28
```

```
    FourthElement <- workersAges[4]
    FourthElement
```

```
## [1] 36
```

```
    SecondFourth <- c(SecondElement, FourthElement)
    SecondFourth
```

```
## [1] 28 36
```

The value in the 2nd element is 28 and in the fourth element is 36.

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

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

```
    x1 <- c("first"=3, "second"=0, "third"=9)
    names(x1)
```

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

```
    x1[c("first", "third")]
```

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

The program output the assigned integer value in the string named “first” and “third” using square brackets [] means accessing through index.

5. Create a sequence x from -3:2.

a. Modify 2nd element and change it to 0

```
    x2 <- -3:2
    x2
```

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

```
    x2[2] = 0
    x2
```

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

The program first produces an output of integers from -3 to 2 in sequence, and when we used an indexing using square brackets [] that access the element value we want to change which is the second position in the index, we modified the original value in that position which is -2 and changed it to 0. Thus, afterwards when prompt for the elements inside the x the second value is now 0.

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.

```
Month = c("Jan", "Feb", "March","April", "May","June")
  PricePerLiter <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)
  PurchaseQuantity <- c(25, 30, 40, 50, 10, 45)

  DieselPurchased <- data.frame (
  Month,
  PricePerLiter,
  PurchaseQuantity
  )
  DieselPurchased
```

```
##   Month PricePerLiter PurchaseQuantity
## 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)

```
Month = c("Jan", "Feb", "March","April", "May","June")
  PricePerLiter <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)
  PurchaseQuantity <- c(25, 30, 40, 50, 10, 45)
 Mean <- weighted.mean(PricePerLiter, PurchaseQuantity)
 Mean
```

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

7. R has actually lots of built-in datasets. 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).

```
    rivers
```

```
##   [1]  735  320  325  392  524  450 1459  135  465  600  330  336  280  315  870
##  [16]  906  202  329  290 1000  600  505 1450  840 1243  890  350  407  286  280
##  [31]  525  720  390  250  327  230  265  850  210  630  260  230  360  730  600
##  [46]  306  390  420  291  710  340  217  281  352  259  250  470  680  570  350
##  [61]  300  560  900  625  332 2348 1171 3710 2315 2533  780  280  410  460  260
##  [76]  255  431  350  760  618  338  981 1306  500  696  605  250  411 1054  735
##  [91]  233  435  490  310  460  383  375 1270  545  445 1885  380  300  380  377
## [106]  425  276  210  800  420  350  360  538 1100 1205  314  237  610  360  540
## [121] 1038  424  310  300  444  301  268  620  215  652  900  525  246  360  529
## [136]  500  720  270  430  671 1770
```

```
    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 displayed a number answers needed with the function lenght, sum, mean, median, var, sd, min, and max used with the elements of rivers.

8. The table below gives the 25 most powerful celebrities and their annual pay as rank.

a. Create vectors according to the above table. Write the codes.

```
PowerRanking <- 1:25
    CelebrityName <- c("Tom Cruise","Rolling Stones", "Oprah Winfrey","U2","Tiger Woods",
                       "Steven Spielberg","Howarf Stern","50 Cent","Cast of the Sopranos","Dan Brown",
                       "Bruce Springsteen", "Donald Trump","Muhammand 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 Bryan")
    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)

CelebrityAnnualPay <- data.frame(
      PowerRanking,
      CelebrityName,
      Pay
    )
    CelebrityAnnualPay
```

```
##    PowerRanking        CelebrityName 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         Howarf 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        Muhammand 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 Bryan  31
```

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

```
CelebrityAnnualPay [19, "PowerRanking"] = 15
      CelebrityAnnualPay[19, "Pay"] = 90
      CelebrityAnnualPay
```

```
##    PowerRanking        CelebrityName 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          Howarf 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        Muhammand 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 Bryan  31
```

c. Interpret the data. The data was accessed and changed by: (1)declaring the object name of the data frame, (2) using square brackets accessing the rank number,(3) by the vector name where the values we want to change, and lastly by declaring the object name again to access the modified data. JK Rowling's rank was changed from 19 to 15 and her annual pay was changed from 75 to 90.