Worksheet-2 in R

KATHRINA CASANDRA D. SISON

**Worksheet for R Programming**

**Instructions:**

* Use RStudio or the RStudio Cloud to accomplish this worksheet.
* Save the R script as Worksheet\_lastname #2.R.
* Commit and push the R script and your Rmarkdown file in html to your own repo. Do not forget to comment your Git repo  
  Accomplish this worksheet by answering the questions being asked and writing the code manually

.

**Using Vectors**

1. Create a vector using : operator
2. Sequence from -5 to 5. Write the R code and its output.  
   Describe its output.

**seq <- c(-5:5)**

**seq**

1. **-5 -4 -3 -2 -1 0 1 2 3 4 5**

**The output creates a sequence starting from -5 and ending at 5, incrementing by 1.**

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

**x <- 1:7**

**x**

**[1] 1 2 3 4 5 6 7**

1. \* Create a vector using seq() function
2. seq(1, 3, by=0.2) # specify step size

Write the R script and its output. Describe the output.

**vec <- seq(1, 3, by = 0.2)**

**print(vec)**

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

The sequence starts at 1.0 and ends at 3.0 each number increases by 2.0.

1. 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.
2. Access 3rd element, what is the value?

third\_element <- ages[3]

print(third\_element)

[1] 22

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

**second\_and\_fourth <- ages[c(2, 4)]**

**print(second\_and\_fourth)**

[1] 28 36

1. Access all but the 4th and 12th element is not included. Write the R script and its output.

**result <- ages[-c(4, 12)]**

**print(result)**

[1] 34 28 22 27 18 52 39 42 29 35 27 22 37 34 19 20 57 49 50 37 46 25 17 37 43 53

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

1. \*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)**

**x**

**first second third**

**3 0 9**

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

Describe the output.

The vector x contains three elements where the value 3 is labeled as "first," the value 0 is labeled as "second," and the value 9 is labeled as "third.". The instruction was to only call the first and third element.

1. Write the code and its output.

**x <- c("first" = 3, "second" = 0, "third" = 9)**

**print(x)**

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

**first third**

**3 9**

1. Create a sequence x from -3:2.
2. Modify 2nd element and change it to 0;

x [2] <- 0

x

Describe the output.

The output shows that the second element of the original vector has been successfully changed from **-2** to **0**, resulting in the updated vector displaying the values -3, 0, -1, 0, 1, and 2.

b.Write the code and its output.

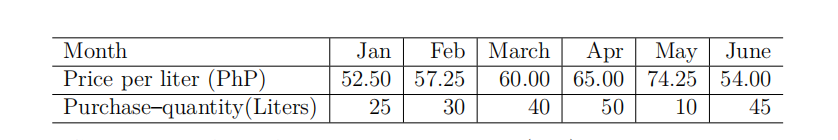
**x <- seq(-3, 2)**

**x[2] <- 0**

**x**

1. **-3 0 -1 0 1 2**

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



1. Create a data frame for month, price per liter (php) and purchase-quantity (liter). Write the R scripts and its output.

**month <- c("Jan", "Feb", "March", "Apr", "May", "June")**

**price\_per\_liter <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)**

**purchase\_quantity <- c(25, 30, 40, 50, 10, 45)**

**diesel\_data <- data.frame(month, price\_per\_liter, purchase\_quantity)**

**print(diesel\_data)**

month price\_per\_liter purchase\_quantity

1 Jan 52.50 25

2 Feb 57.25 30

3 March 60.00 40

4 Apr 65.00 50

5 May 74.25 10

6 June 54.00 45

1. What is the average fuel expenditure of Mr. Cruz from Jan to June? Note: Use 'weighted.mean( liter, purchase); Write the R scripts and its output.

**expenditure <- price\_per\_liter \* purchase\_quantity**

**weighted\_avg\_expenditure <- weighted.mean(price\_per\_liter, purchase\_quantity)**

**print(weighted\_avg\_expenditure)**

**[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).

data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers), var (rivers),

sd(rivers), min(rivers), max(rivers))

1. What are the results?

Length Sum Mean Median

141.0000 83357.0000 591.1844 425.0000

Variance Standard Deviation Minimum Maximum

243908.4086 493.8708 135.0000 3710.0000

C. Write the R scripts and its outputs.

**data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers),**

**var(rivers), sd(rivers), min(rivers), max(rivers))**

**names(data) <- c("Length", "Sum", "Mean", "Median", "Variance", "Standard Deviation", "Minimum", "Maximum")**

**print(data)**

Length Sum Mean Median

141.0000 83357.0000 591.1844 425.0000

Variance Standard Deviation Minimum Maximum

243908.4086 493.8708 135.0000 3710.0000

1. 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.

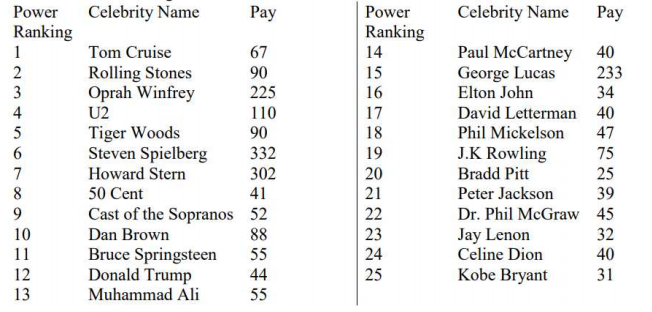


Figure 1: Forbes Ranking

a. Create vectors according to the above table.

Write the R scripts and its output

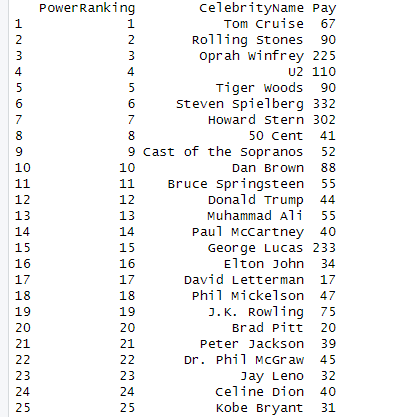
power\_ranking <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)

celebrity\_name <- 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", "Brad Pitt", "Peter Jackson", "Dr. Phil McGraw", "Jay Leno", "Celine Dion", "Kobe Bryant")

pay <- c(67, 90, 225, 110, 90, 332, 302, 41, 52, 88, 55, 44, 55, 40, 233, 34, 17, 47, 75, 20, 39, 45, 32, 40, 31)

forbes\_data <- data.frame(PowerRanking = power\_ranking, CelebrityName = celebrity\_name, Pay = pay)

forbes\_data



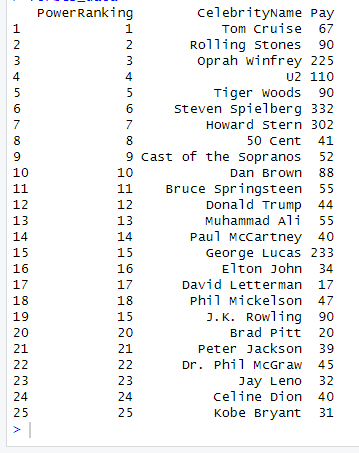
b. Modify the power ranking and pay of J.K. Rowling.

Change power ranking to 15 and pay to 90. Write the R scripts and its output.

forbes\_data[forbes\_data$CelebrityName == "J.K. Rowling", "PowerRanking"] <- 15

forbes\_data[forbes\_data$CelebrityName == "J.K. Rowling", "Pay"] <- 90

forbes\_data

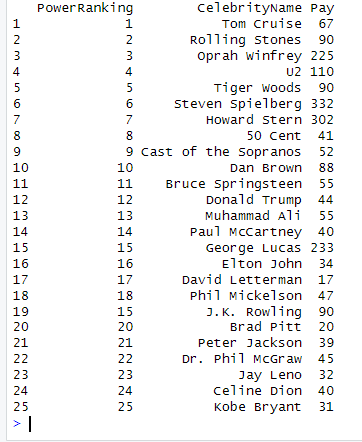


1. Create an excel file from the table above and save it as csv file (PowerRanking). Import the csv file into the RStudio. What is the R script?

write.csv(forbes\_data, file = "PowerRanking.csv", row.names = FALSE)

imported\_data <- read.csv("PowerRanking.csv")

imported\_data



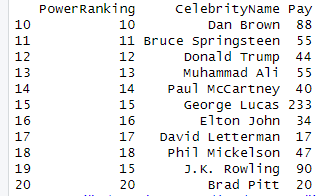
d. Access the rows 10 to 20 and save it as Ranks.RData

Write the R script and its output.

subset\_data <- forbes\_data[10:20, ]

save(subset\_data, file = "Ranks.RData")

subset\_data



1. Describe its output.
2. The output of the forbes\_data data frame presents a structured summary of the power rankings, celebrity names, and their respective earnings.
3. When you execute the commands to modify forbes\_data, you're specifically changing the entries for the celebrity "J.K. Rowling."
4. **It saves the table as a CSV file and imports it back into R, maintaining the data structure.**

**(d)** The subset\_data data frame includes the selected rows (10 to 20) from the original forbes\_data data frame. Each row contains the **PowerRanking**, **CelebrityName**, and **Pay** for the corresponding celebrities, providing a focused view of a specific segment of the original data.

9. Download the Hotels-Vienna https://tinyurl.com/ Hotels- Vienna

a. Import the excel file into your RStudio.

What is the R. script?

**install.packages ("readx]")**

**library(readxl)**

**file\_path <- "C:/Users/Admin/Downloads/hotels-vienna.xlsx"**

**hotels\_data <- read\_excel(file\_path)**

**hotels\_data**

b.How many dimensions does the dataset have?

What is the R script? WHat is its output?

**dataset\_dimensions <- dim(hotels\_data)**

**dataset\_dimensions**

**[1] 428 24**

1. Select columns country, **neighbourhood,**

**price, stars, accommodation\_type,** and

**ratings.** Write the R script.

selected\_data <- hotels\_data[,c("country", "neighbourhood", "price" , "stars", "accommodation\_type", "rating")]

selected\_data

d. Save the data as \*\*new.RData to your RStudio. Write theR. script.

**save (selected\_data, file = "new.RData")**

Display the first six rows and last six rows of the new.RData. What is the R script?

**load("new.RData")**

**head(selected\_data)**

**tail(selected\_data)**

10. Create a list of ten (10) vegetables you ate during your lifetime. If none, just list down.

a. Write the R scripts and its output.

vegetables <- c("potato", "carrot", "squash", "eggplant", "broccoli", "lettuce", "cabbage", "greenbean", "corn", "cucumber")

vegetables

[1] "potato" "carrot" "squash" "eggplant" "broccoli" "lettuce"

[7] "cabbage" "greenbean" "corn" "cucumber"

1. Add 2 additional vegetables after the last vegetables in the list. What is the Rscript and its output.

vegetables <-c(vegetables, "bell pepper", "long beans")

Vegetables

[1] "potato" "carrot" "squash" "eggplant" "broccoli"

[6] "lettuce" "cabbage" "greenbean" "corn" "cucumber"

[11] "bell pepper" "long beans"

1. Add 4 additional vegetables after index 5. How many datapoints does your vegetable list have? What is the R script and its output?

vegetables <- append(vegetables, c("asparagus", "cabbage", "spinach", "celery"), after = 5)

print(vegetables)

num\_datapoints <- length(vegetables)

print(paste("Number of datapoints:", num\_datapoints))

[1] "Number of datapoints: 16"

1. Remove the vegetables in index 5. 10. and 15. How many vegetables were left? Write the codes and its output.

vegetables <- vegetables[-c(5, 10, 15)]

print(vegetables)

num\_vegetables\_left <- length(vegetables)

print(paste("Number of vegetables left:", num\_vegetables\_left))

[1] "Number of vegetables left: 13"

Note: Do not forget to push into your GitHub repo.

Prepared by:

JOYCE F. JAMILE

Instructor

Without ethical considerations, Al becomes a tool of chaos and harm.