

# "WorkSheet2"

**Neil Francis N. Navarro**  
**BSIT 2-A**

## Worksheet for R Programming

### Instructions:

- 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. Write the R code and its output. Describe its output.

```
Seq(-5:5)
```

```
[1] 1 2 3 4 5 6 7 8 9 10 11
```

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

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

```
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.

```
seq(1, 3, by=0.2)
```

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

### Explanation:

The result will shows numbers that is being added by 0.2 from 1 to 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.

```
ages <- 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)
```

```
ages
```

```
[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 37 43 53 41 51 35
[32] 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?**

```
Ages[3]
[1] 22
```

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

```
ages[2]
ages[4]
[1] 28
[1] 36
```

**c. Access all but the 1st element is not included. Write the R code and its output.**

```
ages[2:50]
[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 43 53 41 51 35 24
[32] 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).
names <- c("first"=3, "second"=0, "third"=9)
names
first second third
3 0 9
```

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

```
Describe the output.
names[c("first", "third")]
first third
3 9
```

**b. Write the code and its output.**

```
names[c("first", "third")]
first third
3 9
```

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

```
x <- c(-3:2)
x
```

**a. Modify 2nd element and change it to 0;**

```
x[2] <- 0
x
```

Describe the output.

**Explanation:**

We change the value of the 2<sup>nd</sup> element into 0 that's why the 2<sup>nd</sup> element output is 0.

```
x <- 0
x
[1] -3 -2 -1 0 1 2
```

**b. Write the code and its output.**

```
x <- 0
x
[1] 0
```

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

```
deisel <- data.frame(
  Months = c("Jan", "Feb", "March", "Apr", "May", "June"),
  Php = c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00),
  Liter = c(25, 30, 40, 50, 10, 45),
  stringsAsFactors = FALSE)
deisel
  Months  Php Liter
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
```

**a. Create a data frame for month, price per liter (php) and purchase-quantity (liter). Write the codes.**

```
deisel$Php
[1] 52.50 57.25 60.00 65.00 74.25 54.00
deisel$Liter
[1] 25 30 40 50 10 45
```

**b. What is the average fuel expenditure of Mr. Cruz from Jan to June? Note: Use weighted.mean(Liter,Php)**

```
weighted.mean(Liter,Php)
[1] 32.65152
```

**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))**

```
data <- c(length(rivers), sum(rivers), mean(rivers), median(rivers),
          var(rivers), sd(rivers), min(rivers), max(rivers))
```

**b. What are the results?**

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

**c. Write the code and its outputs.**

```
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 135.0000 3710.0000
```

**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.**

**Figure 1: Forbes Ranking**

```
PowerRanking <- 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)

Ranking <- data.frame(PowerRanking, Celebrities, Pay)
Ranking
```

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

```
PowerRanking <- 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)

Ranking <- data.frame(PowerRanking, Celebrities, Pay)
Ranking
PowerRanking      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.**

```
PowerRanking[19] <- 15;
```

```
PowerRanking
```

```
Pay[19] <- 90;
```

```
Pay
```

```
ArtistRanking <- data.frame(PowerRanking, Celebrities, Pay)
```

```
ArtistRanking
```

	PowerRanking	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.**

```
PowerRanking[19] <- 15;
```

```
PowerRanking
```

```
Pay[19] <- 90;
```

```
Pay
```

```
ArtistRanking <- data.frame(PowerRanking, Celebrities, Pay)
```

```
ArtistRanking
```

PowerRanking	Celebrities	Pay
1	Tom Cruise	67
2	Rolling Stones	90
3	Oprah Winfrey	225
4	U2	110
5	Tiger Woods	90
6	Steven Spielberg	332
7	Howard Stern	302
8	50 Cent	41
9	Cast of the Sopranos	52
10	Dan Brown	88
11	Bruce Springsteen	55
12	Donald Trump	44
13	Muhammad Ali	55
14	Paul McCartney	40
15	George Lucas	233
16	Elton John	34
17	David Letterman	40
18	Phil Mickelson	47
19	J.K Rowling	90
20	Bradd Pitt	25
21	Peter Jackson	39
22	Dr. Phil McGraw	45
23	Jay Lenon	32
24	Celine Dion	40
25	Kobe Bryant	31

**Explanation:**

I have modify the power ranking and pay of J.K. Rowling. And change power ranking to 15 and pay to 90 and this was the output.