

Worksheet #2

Kenan Jake Jimenez

2025-09-29

```
# Worksheet in R Programming
```

```
---
```

```
## Exercise 1. Create a vector using ':' operator
```

```
**a. Sequence from -5 to 5**
```

```
''' r
vec1 <- -5:5
vec1
```

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

Output: [1] -5 -4 -3 -2 -1 0 1 2 3 4 5

The output shows a sequence from -5 up to 5.

```
b. x <- 1:7
```

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

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

Output: [1] 1 2 3 4 5 6 7

Exercise 2. Create a vector using seq() function

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

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

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

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

The numbers increase from 1 to 3 by increments of 0.2.

Exercise 3. Census of Workers

```
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)
```

a. Access 3rd element

```
ages[3]
```

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

Output: [1] 22

b. Access 2nd and 4th elements

```
ages[c(2,4)]
```

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

Output: [1] 28 36

c. Exclude 1st element

```
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
## [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
```

Exercise 4. Named Vector

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

```
## first second third
##      3      0      9
```

a. Access first and third

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

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

Output: first third 3 9

Exercise 5. Modify Sequence

```
seq2 <- -3:2  
seq2
```

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

Output: [1] -3 -2 -1 0 1 2

a. Modify 2nd element \rightarrow 0

```
seq2[2] <- 0  
seq2
```

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

Output: [1] -3 0 -1 0 1 2

Exercise 6. Diesel Fuel Purchased by Mr. Cruz

```
month <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun")  
price <- c(52.50, 57.25, 60.00, 65.00, 74.25, 54.00)  
liters <- c(25, 30, 40, 50, 10, 45)  
  
fuel_data <- data.frame(month, price, liters)  
fuel_data
```

```
##   month price liters  
## 1   Jan  52.50     25  
## 2   Feb  57.25     30  
## 3   Mar  60.00     40  
## 4   Apr  65.00     50  
## 5   May  74.25     10  
## 6   Jun  54.00     45
```

a. Data Frame Output

Month	Price	Liters
Jan	52.50	25
Feb	57.25	30
Mar	60.00	40
Apr	65.00	50
May	74.25	10
Jun	54.00	45

b. Average Fuel Expenditure

```
avg_exp <- weighted.mean(price, liters)
avg_exp
```

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

Output: [1] 59.2625

Exercise 7. Built-in Dataset: rivers

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

```
##      length      sum      mean      median  variance      sd
##  141.0000  83357.0000  591.1844   425.0000 243908.4086  493.8708
##      min      max
##  135.0000  3710.0000
```

Output: length sum mean median variance sd min max 141 591952 591.2 425 98382.96 313.83 135 3710

Exercise 8. Forbes Top Celebrities

```

power_rank <- 1:25
celebrity <- 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, 40, 47, 75, 25,
  39, 45, 32, 40, 31
)

forbes <- data.frame(power_rank, celebrity, pay)
forbes

```

```

##   power_rank      celebrity 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      Brad Pitt  25
## 21        21    Peter Jackson 39
## 22        22   Dr. Phil McGraw 45
## 23        23      Jay Leno  32
## 24        24    Celine Dion  40
## 25        25    Kobe Bryant  31

```

b. Update J.K. Rowling

```

jk_index <- which(forbes$celebrity == "J.K. Rowling")
forbes$power_rank[jk_index] <- 15

```

```
forbes$pay[jk_index] <- 90
forbes[jk_index, ]
```

```
##      power_rank      celebrity pay
## 19           15 J.K. Rowling  90
```

Output: power_rank celebrity pay 15 J.K. Rowling 90

c. Interpretation The output shows celebrities ranked by power and pay. After updating, J.K. Rowling is ranked 15th with a pay of 90 million dollars, putting her among other top earners. This shows how changes in pay can affect overall ranking.
