Rworksheet_Regacho#2

```r
#1. Create a vector using : operator
x <- -5:5
x
```

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

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

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

```r
#2.* Create a vector using seq() function
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
```

```r
#3. A factory has a census of its workers. There are 50 workers in total. The following list shows thei
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)

print(ages[3])
```

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

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

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

```r
print(ages[-1])
```

```
##  [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
```

```r
#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)
x
```

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

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

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

```r
#The Output shows only the first and third which the value is 3 and 9.
```

```r
#5. Create a sequence x from -3:2.
x <- -3:2
x
```

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

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

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

```r
#The output shows that the element [2] was replaced by 0.
```

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

fuel <- data.frame(Month, Priceperliter, Liters)
fuel
```

```
##   Month Priceperliter Liters
## 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
```

```r
ave <- weighted.mean(Liters, Priceperliter)
ave
```

```
## [1] 32.65152
```

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

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

```r
#8.
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)

celeb <- 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 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)

# Combine into a data frame
forbes <- data.frame(ranking, celeb, pay)
print(forbes)
```

```
##    ranking                celeb 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 Leno  32
## 24      24          Celine Dion  40
## 25      25          Kobe Bryant  31
```

```r
### b. Modify J.K. Rowling → ranking = 15, pay = 90
forbes$ranking[forbes$celeb=="J.K Rowling"] <- 15
forbes$pay[forbes$celeb=="J.K Rowling"] <- 90

# Show updated data
```

```r
print(forbes[forbes$celeb=="J.K Rowling" , ])
```

```
##    ranking       celeb pay
## 19      15 J.K Rowling  90
```