

TD1

Contents

Exercise 1	1
Exercise 2 : Opérations avec les tables	2
Exercise 3. Operations with a table of decimal numbers	4
Exercise 4. Graphics	5

Exercise 1

1. Create a vector of number
 - The list from 1 to 100

```
seq(1,100)
```

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
## [18] 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
## [35] 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51
## [52] 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68
## [69] 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85
## [86] 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
```

- A number list/sequence for 10, 20, 25, 50, repeated 5 times (10 20 25 50 10 20 25 50 10 20 25 50 10 20 25 50)

```
rep(c(10,20,25,50),5)
```

```
## [1] 10 20 25 50 10 20 25 50 10 20 25 50 10 20 25 50
```

- a list from 1 to 100, with a step of 5

```
seq(1,100,5)
```

```
## [1] 1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96
```

- Repeat 10 times the number 12

```
rep(12,10)
```

```
## [1] 12 12 12 12 12 12 12 12 12 12
```

2. A vector with the number 1, 2, 3 with a repetition of each 4 times with in addition a repetition of the sequence of 4 times

- Put this vector in the object « VEC »

```
VEC = rep(c(1,2,3),each = 4, times = 4)
```

```
VEC
```

```
## [1] 1 1 1 1 2 2 2 2 3 3 3 3 1 1 1 1 2 2 2 2 3 3 3 3
## [36] 3 1 1 1 1 2 2 2 2 3 3 3 3
```

- Multiply all the element of VEC by 5

```
VEC = VEC * 5
```

```
VEC
```

```
## [1] 5 5 5 5 10 10 10 10 15 15 15 15 5 5 5 5 10 10 10 10 15 15 15
## [24] 15 5 5 5 5 10 10 10 10 15 15 15 15 5 5 5 5 10 10 10 10 15 15
## [47] 15 15
```

- Calculate the median and the quantile of 75%

```
quantile(VEC, 0.5) # Median
```

```
## 50%
```

```
## 10
```

```
quantile(VEC, 0.75)
```

```
## 75%
```

```
## 15
```

3. Create a vector with a serie of number from 1 to 2000 with a step of 10

- Put it in the object « vec »

```
vec = seq(1,2000,10)
```

```
vec
```

```
## [1] 1 11 21 31 41 51 61 71 81 91 101 111 121 131
## [15] 141 151 161 171 181 191 201 211 221 231 241 251 261 271
## [29] 281 291 301 311 321 331 341 351 361 371 381 391 401 411
## [43] 421 431 441 451 461 471 481 491 501 511 521 531 541 551
## [57] 561 571 581 591 601 611 621 631 641 651 661 671 681 691
## [71] 701 711 721 731 741 751 761 771 781 791 801 811 821 831
## [85] 841 851 861 871 881 891 901 911 921 931 941 951 961 971
## [99] 981 991 1001 1011 1021 1031 1041 1051 1061 1071 1081 1091 1101 1111
## [113] 1121 1131 1141 1151 1161 1171 1181 1191 1201 1211 1221 1231 1241 1251
## [127] 1261 1271 1281 1291 1301 1311 1321 1331 1341 1351 1361 1371 1381 1391
## [141] 1401 1411 1421 1431 1441 1451 1461 1471 1481 1491 1501 1511 1521 1531
## [155] 1541 1551 1561 1571 1581 1591 1601 1611 1621 1631 1641 1651 1661 1671
## [169] 1681 1691 1701 1711 1721 1731 1741 1751 1761 1771 1781 1791 1801 1811
## [183] 1821 1831 1841 1851 1861 1871 1881 1891 1901 1911 1921 1931 1941 1951
## [197] 1961 1971 1981 1991
```

- Extract the 10 th value of vec

```
vec[10]
```

```
## [1] 91
```

- Display a sub-vector called « vec2 » that corresponds to the values from 2 to 6

```
vec2 = vec[2:6]
```

```
vec2
```

```
## [1] 11 21 31 41 51
```

- Replace the last value of vec2 by 100

```
vec2[length(vec2)] = 100
```

```
vec2
```

```
## [1] 11 21 31 41 100
```

- Display vec2 without its 3th value and store it in vec3

```
vec3 = vec2[-3]
```

```
vec3
```

```
## [1] 11 21 41 100
```

- Replace all the values ≥ 30 by 30

```
vec3[vec3 >= 30] = 30
```

```
vec3
```

```
## [1] 11 21 30 30
```

Exercice 2 : Opérations avec les tables

Create a matrix with 5 columns and 20 lines with the values from 1 to 100, by lines - Call it MAT

```
MAT = matrix(1:100,byrow = TRUE, ncol = 5)
```

```
MAT
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   10
## [3,]   11   12   13   14   15
## [4,]   16   17   18   19   20
## [5,]   21   22   23   24   25
## [6,]   26   27   28   29   30
## [7,]   31   32   33   34   35
## [8,]   36   37   38   39   40
## [9,]   41   42   43   44   45
## [10,]  46   47   48   49   50
## [11,]  51   52   53   54   55
## [12,]  56   57   58   59   60
## [13,]  61   62   63   64   65
## [14,]  66   67   68   69   70
## [15,]  71   72   73   74   75
## [16,]  76   77   78   79   80
## [17,]  81   82   83   84   85
## [18,]  86   87   88   89   90
## [19,]  91   92   93   94   95
## [20,]  96   97   98   99  100
```

- Display the value at the second line and 5 th column et replace it by NA

```
MAT[2,5] = NA
MAT
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   NA
## [3,]   11   12   13   14   15
## [4,]   16   17   18   19   20
## [5,]   21   22   23   24   25
## [6,]   26   27   28   29   30
## [7,]   31   32   33   34   35
## [8,]   36   37   38   39   40
## [9,]   41   42   43   44   45
## [10,]  46   47   48   49   50
## [11,]  51   52   53   54   55
## [12,]  56   57   58   59   60
## [13,]  61   62   63   64   65
## [14,]  66   67   68   69   70
## [15,]  71   72   73   74   75
## [16,]  76   77   78   79   80
## [17,]  81   82   83   84   85
## [18,]  86   87   88   89   90
## [19,]  91   92   93   94   95
## [20,]  96   97   98   99  100
```

- Create “mat “ with only the columns 2,3 and 4 (taking them from MAT)

```
mat = MAT[,2:4]
mat
```

```
##      [,1] [,2] [,3]
## [1,]    2    3    4
## [2,]    7    8    9
```

```
## [3,] 12 13 14
## [4,] 17 18 19
## [5,] 22 23 24
## [6,] 27 28 29
## [7,] 32 33 34
## [8,] 37 38 39
## [9,] 42 43 44
## [10,] 47 48 49
## [11,] 52 53 54
## [12,] 57 58 59
## [13,] 62 63 64
## [14,] 67 68 69
## [15,] 72 73 74
## [16,] 77 78 79
## [17,] 82 83 84
## [18,] 87 88 89
## [19,] 92 93 94
## [20,] 97 98 99
```

- Replace the values between 40 and 60 by 50 in the matrix « mat »

```
mat[mat>40 && mat<60] = 50
mat
```

```
##      [,1] [,2] [,3]
## [1,]  2   3   4
## [2,]  7   8   9
## [3,] 12  13  14
## [4,] 17  18  19
## [5,] 22  23  24
## [6,] 27  28  29
## [7,] 32  33  34
## [8,] 37  38  39
## [9,] 42  43  44
## [10,] 47  48  49
## [11,] 52  53  54
## [12,] 57  58  59
## [13,] 62  63  64
## [14,] 67  68  69
## [15,] 72  73  74
## [16,] 77  78  79
## [17,] 82  83  84
## [18,] 87  88  89
## [19,] 92  93  94
## [20,] 97  98  99
```

Exercise 3. Operations with a table of decimal numbers

1. Get the file (moodle) M&Ms.xls and open it with excel or open-office
 - save it as MetMs.txt
 - Import it in R and store it in data

```
library(readxl)
data <- read_excel("~/Téléchargements/MetMs.xlsx")
```

2. Create data2 by removing the variable largeur (width) from MetMs and the missing data

```
data2 = subset(data, select = -Forme)
data2 = na.omit(data2)
data2
```

```
## # A tibble: 100 x 4
##   Indice Couleur `Longueur maximale (mm)` `Poids (g)`
##   <dbl> <chr>          <dbl>          <dbl>
## 1      1  Orange          17.4            2
## 2      2   Bleu          20.9           2.4
## 3      3   Vert          15.1           1.9
## 4      4   Jaune          15.4            2
## 5      5   Bleu          15.6           1.7
## 6      6  Marron          17.6           2.9
## 7      7  Marron          14.2           1.7
## 8      8   Bleu          15.7           2.3
## 9      9  Marron          17.3           2.4
## 10     10  Jaune          16.9           2.5
## # ... with 90 more rows
```

- Determine for which colors the lengths are minimal and maximal

```
with(data2, Couleur[`Longueur maximale (mm)` == min(`Longueur maximale (mm)`)])
```

```
## [1] "Bleu"
```

```
with(data2, Couleur[`Longueur maximale (mm)` == max(`Longueur maximale (mm)`)])
```

```
## [1] "Rouge"
```

- Sort the lines of data2 according to an increasing order of the length

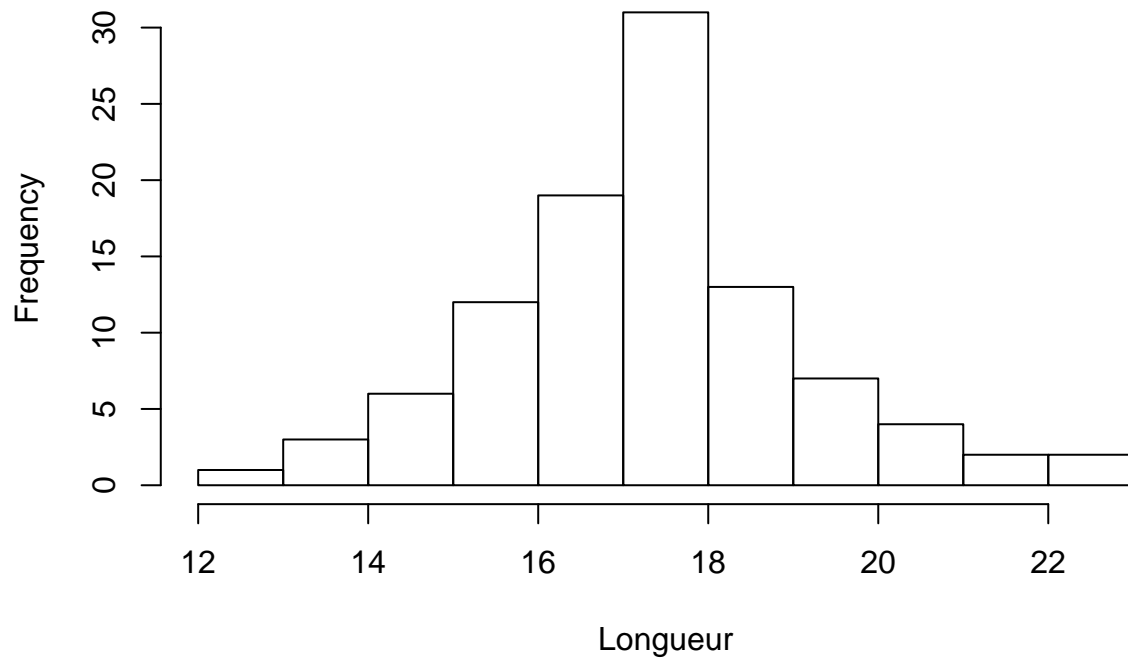
```
data2[order(data2$`Longueur maximale (mm)`),]
```

```
## # A tibble: 100 x 4
##   Indice Couleur `Longueur maximale (mm)` `Poids (g)`
##   <dbl> <chr>          <dbl>          <dbl>
## 1     13  Bleu          12.8           1.2
## 2     99  Bleu          13.4           1.2
## 3     60  Vert          13.7           1.7
## 4     67  Jaune          14            1.5
## 5      7  Marron          14.2           1.7
## 6     87  Vert          14.5           1.6
## 7     93  Orange          14.5           1.8
## 8     11  Rouge          14.6           1.9
## 9     97  Bleu          14.9           1.7
## 10    92  Orange          15            1.9
## # ... with 90 more rows
```

Exercice 4. Graphics

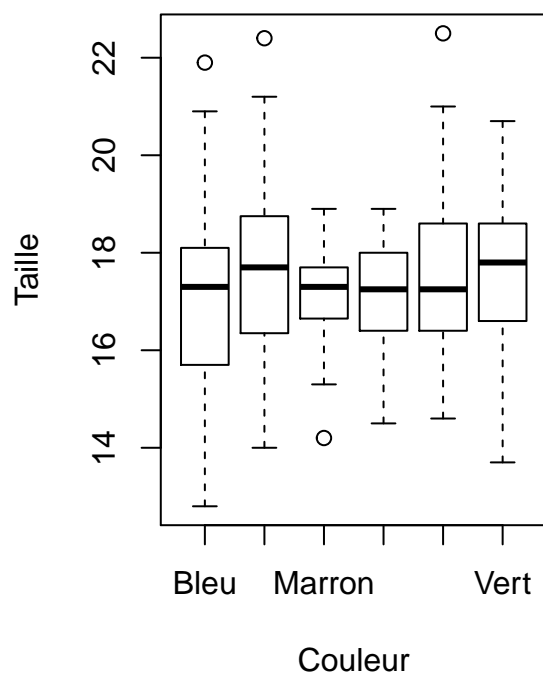
```
hist(data2$`Longueur maximale (mm)`, xlab = "Longueur", main = "Longueur des M & M's")
```

Longueur des M & M's



```
par(mfrow=c(1,2))
boxplot(data = data2,`Longueur maximale (mm)`~ Couleur ,
        main = "Longueur des M & M's par couleur",
        xlab= "Couleur", ylab= "Taille")
boxplot(data = data2,`Poids (g)`~ Couleur ,
        main = "Poid des M & M's par couleur",
        xlab= "Couleur", ylab= "Poid")
```

Longueur des M & M's par couleur



Poid des M & M's par couleur

