

# Exercise #1

Quillo

2024-03-06

A. Load the built-in warpbreaks dataset.

```
data("warpbreaks")
```

1. Find out, in a single command, which columns of warpbreaks are either numeric or integer. What are the data types of each column?

```
test <- str(warpbreaks)
```

```
## 'data.frame':  54 obs. of  3 variables:
## $ breaks : num  26 30 54 25 70 52 51 26 67 18 ...
## $ wool   : Factor w/ 2 levels "A","B": 1 1 1 1 1 1 1 1 1 1 ...
## $ tension: Factor w/ 3 levels "L","M","H": 1 1 1 1 1 1 1 1 1 2 ...
```

```
test
```

```
## NULL
```

2. How many observations does it have?

```
obs <- objects(warpbreaks)
obs2 <- nrow(warpbreaks)
print(paste(obs, obs2))
```

```
## [1] "breaks 54" "tension 54" "wool 54"
```

3. Is numeric a natural data type for the columns which are stored as such? Convert to integer when necessary.

```
warpbreaks$breaks <- warpbreaks
warpbreaks$wool <- warpbreaks
warpbreaks$tension <- warpbreaks

#intwarp1 <- as.integer(intwarp$breaks)
#intwarp2 <- as.integer(intwarp$wool)
#intwarp3 <- as.integer(intwarp$tension)
```

4. Error messages in R sometimes report the underlying type of an object rather than the user-level class. Derive from the following code and error message what the underlying type. Explain what is the error all about. Do not just copy the error message that was displayed.

```
# The error was on the type of object. It said that its a list and cant be converted to integer.
```

B. Load the exampleFile.txt 1. Read the complete file using readLines.

```
exFile <- readLines("exampleFile.txt")
```

```
## Warning in readLines("exampleFile.txt"): incomplete final line found on
## 'exampleFile.txt'
```

2. Separate the vector of lines into a vector containing comments and a vector containing the data. Hint: use `grepl`.

```
newexFile4 <- exFile[grepl("//",exFile)]

newexFile41 <- exFile[!grepl("//",exFile)]
newexFile41
```

```
## [1] "M;28;81.3"      "male;45;"      "Female;17;57,2" "fem.;64;62.8"
```

3. Extract the date from the first comment line and display on the screen “It was created data.”

```
date <- exFile[1]

line1 <- "It was created"
line2 <- "data."
print(paste(line1, date, line2))
```

```
## [1] "It was created // Survey data. Created : 21 May 2013 data."
```

4. Read the data into a matrix as follows.
  - a. Split the character vectors in the vector containing data lines by semicolon (;) using `strsplit`.

```
hehe <- exFile
newexFile3 <- strsplit(hehe, ';')
newexFile3
```

```
## [[1]]
## [1] "// Survey data. Created : 21 May 2013"
##
## [[2]]
## [1] "// Field 1: Gender"
##
## [[3]]
## [1] "// Field 2: Age (in years)"
##
## [[4]]
## [1] "// Field 3: Weight (in kg)"
##
## [[5]]
## [1] "M"      "28"      "81.3"
##
## [[6]]
## [1] "male" "45"
##
## [[7]]
## [1] "Female" "17"      "57,2"
##
## [[8]]
## [1] "fem." "64"      "62.8"
```

- b. Find the maximum number of fields retrieved by split. Append rows that are shorter with NA's.

```
(length <- sapply(newexFile3, length))
```

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

```
length
```

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

```

(maxx <- max(length))
## [1] 3
maxx
## [1] 3
(newexFile3 <- lapply(newexFile3, function(row) c(row, rep(NA, maxx - length(row)))))
## [[1]]
## [1] "// Survey data. Created : 21 May 2013"
## [2] NA
## [3] NA
##
## [[2]]
## [1] "// Field 1: Gender" NA NA
##
## [[3]]
## [1] "// Field 2: Age (in years)" NA
## [3] NA
##
## [[4]]
## [1] "// Field 3: Weight (in kg)" NA
## [3] NA
##
## [[5]]
## [1] "M" "28" "81.3"
##
## [[6]]
## [1] "male" "45" NA
##
## [[7]]
## [1] "Female" "17" "57,2"
##
## [[8]]
## [1] "fem." "64" "62.8"
newexFile3
## [[1]]
## [1] "// Survey data. Created : 21 May 2013"
## [2] NA
## [3] NA
##
## [[2]]
## [1] "// Field 1: Gender" NA NA
##
## [[3]]
## [1] "// Field 2: Age (in years)" NA
## [3] NA
##
## [[4]]
## [1] "// Field 3: Weight (in kg)" NA
## [3] NA
##
## [[5]]
## [1] "M" "28" "81.3"

```

```
##
## [[6]]
## [1] "male" "45"  NA
##
## [[7]]
## [1] "Female" "17"      "57,2"
##
## [[8]]
## [1] "fem." "64"    "62.8"
c. Use unlist and matrix to transform the data to row-column format.
```

```
(unlist1 <- unlist(newexFile3))
## [1] "// Survey data. Created : 21 May 2013"
## [2] NA
## [3] NA
## [4] "// Field 1: Gender"
## [5] NA
## [6] NA
## [7] "// Field 2: Age (in years)"
## [8] NA
## [9] NA
## [10] "// Field 3: Weight (in kg)"
## [11] NA
## [12] NA
## [13] "M"
## [14] "28"
## [15] "81.3"
## [16] "male"
## [17] "45"
## [18] NA
## [19] "Female"
## [20] "17"
## [21] "57,2"
## [22] "fem."
## [23] "64"
## [24] "62.8"
```

```
unlist1
## [1] "// Survey data. Created : 21 May 2013"
## [2] NA
## [3] NA
## [4] "// Field 1: Gender"
## [5] NA
## [6] NA
## [7] "// Field 2: Age (in years)"
## [8] NA
## [9] NA
## [10] "// Field 3: Weight (in kg)"
## [11] NA
## [12] NA
## [13] "M"
## [14] "28"
## [15] "81.3"
## [16] "male"
## [17] "45"
```

```

## [18] NA
## [19] "Female"
## [20] "17"
## [21] "57,2"
## [22] "fem."
## [23] "64"
## [24] "62.8"

(mat1 <- matrix(unlist1, nrow = length(data), byrow = TRUE))

##      [,1] [,2] [,3] [,4]
## [1,] "// Survey data. Created : 21 May 2013" NA NA "// Field 1: Gender"
##      [,5] [,6] [,7] [,8] [,9]
## [1,] NA NA "// Field 2: Age (in years)" NA NA
##      [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17]
## [1,] "// Field 3: Weight (in kg)" NA NA "M" "28" "81.3" "male" "45"
##      [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## [1,] NA "Female" "17" "57,2" "fem." "64" "62.8"

mat1

##      [,1] [,2] [,3] [,4]
## [1,] "// Survey data. Created : 21 May 2013" NA NA "// Field 1: Gender"
##      [,5] [,6] [,7] [,8] [,9]
## [1,] NA NA "// Field 2: Age (in years)" NA NA
##      [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17]
## [1,] "// Field 3: Weight (in kg)" NA NA "M" "28" "81.3" "male" "45"
##      [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## [1,] NA "Female" "17" "57,2" "fem." "64" "62.8"

d. From comment lines 2-4, extract the names of the fields. Set these as
   colnames for the matrix you just created.

```r
(namess <- gsub(".*: ", "", newexFile41[2:4]))
## [1] "male;45;" "Female;17;57,2" "fem.;64;62.8"

namess

## [1] "male;45;" "Female;17;57,2" "fem.;64;62.8"

```