

71

## 72 # Ενσωματωμένες συναρτήσεις (build-in functions) του R -----

```
73 read.table(file, header = FALSE, sep = "", quote = "\"",  
74           dec = ".", numerals = c("allow.loss", "warn.loss",  
75           "no.loss"),  
76           row.names, col.names, as.is = !stringsAsFactors,  
77           na.strings = "NA", colClasses = NA, nrow = -1,  
78           skip = 0, check.names = TRUE, fill = !blank.lines.skip,  
79           strip.white = FALSE, blank.lines.skip = TRUE,  
80           comment.char = "#",  
81           allowEscapes = FALSE, flush = FALSE,  
82           stringsAsFactors = default.stringsAsFactors(),  
83           fileEncoding = "", encoding = "unknown", text, skipNul =  
84           FALSE)
```

83

```
84 read.csv(file, header = TRUE, sep = ",", quote = "\"",  
85          dec = ".", fill = TRUE, comment.char = "", ...)
```

86

```
87 read.csv2(file, header = TRUE, sep = ";", quote = "\"",  
88           dec = ",", fill = TRUE, comment.char = "", ...)
```

89

```
90 read.delim(file, header = TRUE, sep = "\t", quote = "\"",  
91           dec = ".", fill = TRUE, comment.char = "", ...)
```

92

```
93 read.delim2(file, header = TRUE, sep = "\t", quote = "\"",  
94            dec = ",", fill = TRUE, comment.char = "", ...)
```

95

```
96 scan(file = "", what = double(), nmax = -1, n = -1, sep = "",  
97      quote = if(identical(sep, "\n")) "" else "\"", dec = ".",  
98      skip = 0, nlines = 0, na.strings = "NA",  
99      flush = FALSE, fill = FALSE, strip.white = FALSE,  
100     quiet = FALSE, blank.lines.skip = TRUE, multi.line = TRUE,  
101     comment.char = "", allowEscapes = FALSE,  
102     fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)
```

103

## 104 # Συναρτήσεις από το πακέτο gdata του R -----

105

```
106 read.xls(xls, sheet=1, verbose=FALSE, pattern,  
107          na.strings=c("NA", "#DIV/0!"),  
108          ..., method=c("csv", "tsv", "tab"), perl="perl")
```

108

```
109 xls2csv(xls, sheet=1, verbose=FALSE, blank.lines.skip=TRUE, ...,
```

109

```

perl="perl")
110
111 xls2tab(xls, sheet=1, verbose=FALSE, blank.lines.skip=TRUE, ...,
perl="perl")
112
113 xls2tsv(xls, sheet=1, verbose=FALSE, blank.lines.skip=TRUE, ...,
perl="perl")
114
115 xls2sep(xls, sheet=1, verbose=FALSE, blank.lines.skip=TRUE, ...,
116         method=c("csv", "tsv", "tab"), perl="perl")
117
118
119 # Συναρτήσεις από το πακέτο readr του R -----
120 # Files ending in .gz, .bz2, .xz, or .zip
121 #   will be automatically uncompressed
122
123 read_delim(file, delim, quote = "\"", escape_backslash = FALSE,
124            escape_double = TRUE, col_names = TRUE, col_types = NULL,
125            locale = default_locale(), na = c("", "NA"), quoted_na = TRUE,
126            comment = "", trim_ws = FALSE, skip = 0, n_max = Inf,
127            guess_max = min(1000, n_max), progress = show_progress())
128
129 read_csv(file, col_names = TRUE, col_types = NULL,
130          locale = default_locale(), na = c("", "NA"), quoted_na = TRUE,
131          quote = "\"", comment = "", trim_ws = TRUE, skip = 0, n_max = Inf,
132          guess_max = min(1000, n_max), progress = show_progress())
133
134 read_csv2(file, col_names = TRUE, col_types = NULL,
135           locale = default_locale(), na = c("", "NA"), quoted_na = TRUE,
136           quote = "\"", comment = "", trim_ws = TRUE, skip = 0, n_max = Inf,
137           guess_max = min(1000, n_max), progress = show_progress())
138
139 read_tsv(file, col_names = TRUE, col_types = NULL,
140          locale = default_locale(), na = c("", "NA"), quoted_na = TRUE,
141          quote = "\"", comment = "", trim_ws = TRUE, skip = 0, n_max = Inf,
142          guess_max = min(1000, n_max), progress = show_progress())
143
144
145 read_fwf(file, col_positions, col_types = NULL, locale =
default_locale(),
146          na = c("", "NA"), comment = "", skip = 0, n_max = Inf,
147          guess_max = min(n_max, 1000), progress = show_progress())

```

```
148
149 read_table(file, col_names = TRUE, col_types = NULL,
150   locale = default_locale(), na = "NA", skip = 0, n_max = Inf,
151   guess_max = min(n_max, 1000), progress = show_progress(), comment
152   = "")
153
154 read_table2(file, col_names = TRUE, col_types = NULL,
155   locale = default_locale(), na = "NA", skip = 0, n_max = Inf,
156   guess_max = min(n_max, 1000), progress = show_progress(), comment
157   = "")
158
159 read_log(file, col_names = FALSE, col_types = NULL, skip = 0,
160   n_max = Inf, progress = show_progress())
161
162 # Συναρτήσεις από το πακέτο readxl του R -----
163
164 read_excel(path, sheet = NULL, range = NULL, col_names = TRUE,
165   col_types = NULL, na = "", trim_ws = TRUE, skip = 0, n_max = Inf,
166   guess_max = min(1000, n_max))
```

```
167 read_xls(path, sheet = NULL, range = NULL, col_names = TRUE,
168   col_types = NULL, na = "", trim_ws = TRUE, skip = 0, n_max = Inf,
169   guess_max = min(1000, n_max))
170
171 read_xlsx(path, sheet = NULL, range = NULL, col_names = TRUE,
172   col_types = NULL, na = "", trim_ws = TRUE, skip = 0, n_max = Inf,
173   guess_max = min(1000, n_max))
174
175 excel_sheets(readxl_example("datasets.xlsx"))
176 excel_sheets(readxl_example("datasets.xls"))
177
178 # To load all sheets in a workbook, use lapply
179 path <- readxl_example("datasets.xls")
180 lapply(excel_sheets(path), read_excel, path = path)
```

```
181
182 # Συναρτήσεις από το πακέτο xlsx του R -----
183
184
185 read.xlsx(file, sheetIndex, sheetName=NULL, rowIndex=NULL,
186   startRow=NULL, endRow=NULL, colIndex=NULL, as.data.frame=TRUE,
187   header=TRUE, colClasses=NA, keepFormulas=FALSE, encoding="unknown",
```

```
...)  
186  
187 read.xlsx2(file, sheetIndex, sheetName=NULL, startRow=1,  
colIndex=NULL, endRow=NULL, as.data.frame=TRUE, header=TRUE,  
colClasses="character", ...)  
188  
189 write.xlsx(x, file, sheetName="Sheet1", col.names=TRUE,  
row.names=TRUE, append=FALSE)  
190  
191 write.xlsx2(x, file, sheetName="Sheet1", col.names=TRUE,  
row.names=TRUE, append=FALSE)
```

## 194 # Συναρτήσεις από το πακέτο openxlsx του R -----

```
195  
196 read.xlsx(xlsxFile, sheet = 1, startRow = 1, colNames = TRUE,  
197 rowNames = FALSE, detectDates = FALSE, skipEmptyRows = TRUE,  
198 skipEmptyCols = TRUE, rows = NULL, cols = NULL, check.names = FALSE,  
199 namedRegion = NULL, na.strings = "NA", fillMergedCells = FALSE)  
200  
201 write.xlsx(x, file, asTable = FALSE, ...)
```

## 204 # Συναρτήσεις από το πακέτο XLConnect του R -----

```
205  
206 loadWorkbook(filename, create = FALSE, password = NULL)  
207 createSheet ( object , name )  
208
```

```

2
3 ## Download a file from the Internet
4 download.file(url, destfile, method, quiet = FALSE, mode = "w",
5               cacheOK = TRUE,
6               extra = getOption("download.file.extra"), ...)
7
8 # Extract files from or list a zip archive
9 unzip(zipfile, files = NULL, list = FALSE, overwrite = TRUE,
10        junkpaths = FALSE, exdir = ".", unzip = "internal",
11        setTimes = FALSE)
12
13 ## Data from *.csv (copy-and-paste) -----
14 # Select the table from an excel file, copy, go to the R Console and
15 # type (past from the clipboard):
16 mydata <- read.table("clipboard", header=TRUE, sep="\t")
17
18 ## Data from *.csv -----
19 # Reading the data directly
20 mydata <- read.csv("c:\mydata\mydatafile.csv", header=TRUE)
21 # The will open a window to search for the *.csv file.
22 mydata <- read.csv(file.choose(), header = TRUE)
23
24 ## Data from/to *.txt (space , tab, comma-separated) -----
25 # In the event that, e.g. the variables have spaces and missing data is coded, e.g. as '-9'
26 mydata <- read.table(("C:/myfolder/abc.txt",
27                      header=TRUE, sep="\t", na.strings = "-9")
28 # Export the data
29 write.table(mydata, file = "test.txt", sep = "\t")
30
31 ## Data from ASCII Record form
32 mydata.dat <-
33 read.fwf(file="http://dss.princeton.edu/training/mydata.dat",
34          width=c(7, -16, 2, 2, -4, 2, -10, 2, -110, 3, -6, 2),
35          col.names=c("w", "y", "x1", "x2", "x3", "age", "gender"), n=1090)
36 # Reading ASCII record form, numbers represent the width of variables,
37 # negative sign excludes variables not wanted (you must include these).
38 # To get the width of the variables you must have a codebook for the
39 # data set available (see an example below).
40 # To get the widths for unwanted spaces use the
41 # formula: Start of var(t+1) – End of var(t) - 1
42
43 # change the names of the columns in a data frame

```

```
44 names(data) <- c("new_name", "another_new_name")
45 # change the name of a column using the name to identify which column to apply the
   name to
46 #   Rename a column in R
47 colnames(data)[colnames(data)=="old_name"] <- "new_name"
48 # Rename a column in R using the column number to identify
49 #   which column to apply the name to
50 names(data)[3]<-"new_name"
51
52
53 # Exploring the data
54 str(mydata) # Provides the structure of the dataset
55 summary(mydata) # Provides basic descriptive statistics and frequencies
56 names(mydata) # Lists variables in the dataset
57 head(mydata) # First 6 rows of dataset
58 head(mydata, n=10)# First 10 (or any other number of) rows of dataset
59 head(mydata, n= -10) # All rows but the last 10
60 tail(mydata) # Last 6 rows
61 tail(mydata, n=10) # Last 10 rows (or any other number of) last rows of dataset
62 tail(mydata, n= -10) # All rows but the first 10
63 mydata[1:10, ] # First 10 rows of the
```

```
64 mydata[1:10,1:3] # First 10 rows of data of the first 3 variables
65 edit(mydata) # Open data editor
66
67
68 # Check for missing data
69 sum(is.na(mydata))# Number of missing in dataset
70 rowSums(is.na(data))# Number of missing per variable
71 rowMeans(is.na(data))*length(data)# No. of missing per row
72 mydata[mydata$age=="& ", "age"] <- NA # NOTE: Notice hidden spaces.
73 mydata[mydata$age=="999", "age"] <- NA
74
75 The function complete.cases() returns a logical vector
76 indicating which cases are complete.
77
78 # list rows of data that have missing values
79 mydata[!complete.cases(mydata), ]
80
81 The function na.omit() returns the object with listwise
82 deletion of missing values.
83
84 # create new dataset without missing data
```

```
85 newdata <- na.omit(mydata)
```

```
86
```