Solutions to Exercises from Chapter 4

1.1-. The three main R functions to use to import data from an ASCII text file are: `read.table()`, `scan()` and `read.ftable()`.

1.2-. `header`: a logical value indicating whether the file contains the names of the variables as its first line (e.g. : `header=TRUE`).
`sep`: the field separator character. Values on each line of the file are separated by this character (e.g.: `sep="  "` or `sep="\t"`).
`dec`: the character used in the file for decimal points (e.g.: `dec="."` or `dec=","`).
`row.names`: a vector of row names. This can be a vector giving the actual row names, or a single number giving the column of the table which contains the row names (e.g.: `row.names=2`).
`skip`: the number of lines of the data file to skip before beginning to read data (e.g.: `skip=4` to exclude the first 4 lines from reading).
`nrows`: the maximum number of rows to read in (e.g.: `row.names=19`).

1.3-. Function `readLines()` reads some or all text lines from a connection.

1.4-. Function `fix()` enables one to modify a `data.frame` or a matrix using a small spreadsheet.

1.5-. `read.csv()`: reads a comma separated value file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file (note: `dec="."`).
`read.csv2()`: reads a semicolon separated value file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file (note: `dec=","`).
`read.delim()`: reads a tabulated separated value file (note: `dec="."`).
`read.delim2()`: reads a tabulated separated value file (note: `dec=","`).

1.6-. Function `read.ftable()` reads, writes and coerces flat contingency tables.

1.7-. Function `scan()` should be used when data are not organised in table format. Function `read.table()` is used for table format data sets.

1.8-. Importing data from an Excel sheet:
• Using copy-paste: select the data under Excel, copy these data to the clipboard, use the instruction:

```
x <- read.table(file("clipboard"),sep="\t",header=TRUE,dec=",")
```

- Using an intermediate ASCII file: save the Excel sheet as `.txt` (separator: TAB), then use function `read.table()`.
- Using package `gdata`gdata and function `read.xls()`.

1.9-. Package `foreign`foreign.

1.10-. The `colClasses` argument from function `read.table()` can be used to indicate the type of each column, and thus greatly increases the speed of reading of huge data sets.

1.11-. Function `write.table()` enables one to write in a file the data set contained in a `data.frame`. Another function is `write()` that should be used for vector or matrix objects.

1.12-. Here are four basic functions to build vectors:

- `c()`
- `seq()`
- `rep()`
- `":"()` (example `1:10`)

1.13-. The instruction `seq(1,2,by=0.1)` gives the following vector:

```
[1] 1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0
```

1.14-. The instruction `rep(1:3,each=2)` gives the following vector:

```
1 1 2 2 3 3
```

1.15-. The instruction `rep(1:3,2)` gives the following vector:

```
1 2 3 1 2 3
```

1.16-. Functions to enter data at hand in a small spread sheet are: `data.entry()` and `de()`.