Chapter 4:
Importing into R

Text Files

✓ read.table Function

➡ To load the contents of a text file into R, the function *read.table*() may be used

- Usage: Type *read.table*({file}, skip = , sep =, header = ,…) at the command line
- Example: For file "testdata.dat" with the following data:

```
Filename: Testdata.dat
Date of creation: Aug 30, 2012
Purpose: To test loading data into R
Show, LeadCharacter1, LeadCharacter2, SuppCharacter2, SuppCharacter2,
NumYears
Seinfeld, Jerry, George, Elaine, Kramer, 15
Big Bang Theory, Sheldon, Leonard, Koothrapalli, Wolowitz, 6
Friends, Chandler, Joey, Ross, Monica, 12
```

- Set the working directory to the file path
- Assign the file to a variable, say *tdata*
- Type mytdata <- read.table(data, skip=3, sep="","", header=TRUE) will yield:

```
> mytdata
             Show LeadCharacter1 LeadCharacter2 SuppCharacter2 SuppCharacter2.1 NumYears
1        Seinfeld          Jerry         George         Elaine           Kramer       15
2 Big Bang Theory        Sheldon        Leonard   Koothrapalli         Wolowitz        6
3         Friends       Chandler           Joey           Ross           Monica       12
```

Text Files

- ✓ read.table Function
  - ➡ Useful arguments
    - stringsAsFactors: This allows for character columns to not be stored as Factors if desired. Possible values: True or False. Overridden by colClasses.
    - colClasses: Used to assign data types to columns. If unused, R will store all columns as characters. Usage: colClasses=c("numeric", "character", "factor",…)
    - quote: Specifies the quote character. If used, any value within quotes, as specified by using the quote character, will be loaded as is. Usage: quote= "{character}"
    - nrows: Used to specify the maximum number of rows to be loaded.
    - col.names: Used to specify a vector of columns names.

- ✓ Variants of *read.table*()
  - ➡ Function read.csv() is used to read csv files
  - ➡ read.fwf() is used to read text files of fixed width format
  - ➡ read.delim() is used to read tab-delimited files

Text Files

✓ scan Function

➡ Reads all data in a file into a single vector of a fixed data type

- Usage: Type *scan*({file}, what= , sep= skip= , …) at the command line
- Example: For file "testdata.dat" with the following data:

```
Filename: Testdata.dat
Date of creation: Aug 30, 2012
Purpose: To test loading data into R
Show, LeadCharacter1, LeadCharacter2, SuppCharacter2, SuppCharacter2,
NumYears
Seinfeld, Jerry, George, Elaine, Kramer, 15
Big Bang Theory, Sheldon, Leonard, Koothrapalli, Wolowitz, 6
Friends, Chandler, Joey, Ross, Monica, 12
```

- Set the working directory to the file path
- Assign the file to a variable, say *tdata*
- Type mytdata <- scan(data, what= "character", sep= ",", skip=3) will yield:

```
> mytdata
 [1] "Show"           " LeadCharacter1" " LeadCharacter2" " SuppCharacter2" " SuppCharacter2"
 [6] " NumYears"      "Seinfeld"        " Jerry"          " George"         " Elaine"
[11] " Kramer"        " 15"             "Big Bang Theory" " Sheldon"       " Leonard"
[16] " Koothrapalli"  " Wolowitz"       " 6"              "Friends"         " Chandler"
[21] " Joey"          " Ross"           " Monica"         " 12 "
```

Exercise

Excel Files

✓ read.xlsx Function

➡ To load the contents of an Excel file into R, the function *read.xlsx*() in package "xlsx" may be used

- Usage: Type *read.xlsx*({file}, sheetindex = , rowIndex =, colIndex =, as.data.frame =, header =, keepFormulas =,…) at the command line
- Example: For file "phones.xls" with the following data:

	A	B	C	D	E	F	G	H
1	Phone	Maker	Price	Country	No_Sold	OperSys	No_Apps	Carrier
2	iPhone	Apple	399	USA	2687161	iOS	3000000	AT&T
3	Galaxy	Samsung	350	Korea	256121	Android	5716247	Verizon
4	Razr	Motorola	200	USA	26511	Android	12381	Sprint
5	Pearl	Blackberry	399	Canada	125819	Blackberry	123701	Rogers
6	Optimus One	LG	299	Korea	123291	Android	12312	AT&T
7	Lumia 800	Nokia	299	Finland	23432	Microsoft	87699	Verizon

- Load package "*xlsx*"
- Set the working directory to the file path
- Perform the steps below

```
> x <- read.xlsx("phones.xls",1)
> x
        Phone     X.Maker X.Price X.Country X.No_Sold   X.OperSys X.No_Apps X.Carrier
1      iPhone       Apple     399       USA   2687161         iOS   3000000      AT&T
2      Galaxy     Samsung     350     Korea    256121     Android   5716247   Verizon
3        Razr    Motorola     200       USA     26511     Android     12381    Sprint
4       Pearl  Blackberry     399    Canada    125819  Blackberry    123701    Rogers
5 Optimus One          LG     299     Korea    123291     Android     12312      AT&T
6   Lumia 800       Nokia     299   Finland     23432   Microsoft     87699   Verizon
> typeof(x)
[1] "list"
> class(x)
[1] "data.frame"
```

Excel Files

- ✓ read.xlsx Function
  - ➡ Useful arguments
    - sheetIndex: Specifies the sheet# in the workbook.
    - rowIndex: A vector specifying the required rows.
    - colIndex: A vector specifying the required columns
    - colClasses: Used to assign data types to columns. If unused, R will store all columns as characters. Usage: colClasses=c("numeric","character","factor",…) colIndex: A vector specifying the required columns
    - as.data.frame: If TRUE, the worksheet is stored as a data frame. If FALSE, it is loaded as a list with one element for each column
    - header: Specifies whether the first row has the column headers
    - keepFormulas: If TRUE, formulas are stored as text in R and not evaluated before import.

- ✓ Variants of *read.xlsx*()
  - ➡ Function read.xlsx2() may be used for larger volume data, given it does more work in Java

Exercise