



Các phương thức xử lý dữ liệu

Đến từ trường đại học

đến từ trường đại

## R for Data Science

### Bài 9: Text, CSV, Excel

Phòng LT & Mạng

[https://csc.edu.vn/lap-trinh-va-cSDL/R-Programming-Language-for-Data-Science\\_190](https://csc.edu.vn/lap-trinh-va-cSDL/R-Programming-Language-for-Data-Science_190)

2020



## Nội dung

1. Text file
2. Csv file
3. Excel file



## Text file

---

### □ Làm việc với thư mục

- Lấy và in thư mục hiện hành `getwd()`

```
> # Get and print current working directory.
> print(getwd())
[1] "C:/Users/KTPHUONG/Documents"
```

- Thay đổi thư mục hiện hành  
`setwd(đường_dẫn)`

```
> # Set current working directory.
> setwd("~/R/chuong11")
>
> # Get and print current working directory.
> print(getwd())
[1] "C:/Users/KTPHUONG/Documents/R/chuong11"
```



## Text file

---

### □ Đọc file text

- Sử dụng `readLines("Đường dẫn tập tin")`

- Ví dụ:

```
> content <- readLines("~/R/chuong11/Du_lieu/license.txt")
> print(content)
[1] "These data files are derived by splitting up the data set downloaded from:"
[2] "https://www.ontario.ca/data/ontario-public-schools-enrolment"
[3] "under the Open Government License - Ontario:"
[4] "https://www.ontario.ca/page/open-government-licence-ontario"
```

- Với tập tin tiếng Việt có dấu, thêm thuộc tính `encoding="UTF-8"`

```
> content <- readLines("~/R/chuong11/Du_lieu/bai_tho.txt", encoding = "UTF-8")
> print(content)
[1] "Mùa xuân là Tết trồng cây"                                "Làm cho đất nước càng ngày càng xuân"
[3] ""
```



## Text file

---

### ❑ Các phương thức:

- Đếm số lượng element trong vector kết quả đọc từ tập tin: **length()**
- Đếm số ký tự của từng element trong vector kết quả đọc từ tập tin: **nchar()**

#### ▪ Ví dụ

```
> content1 <- readLines("~/R/chuong11/Du_lieu/license.txt")
> print(content1)
[1] "These data files are derived by splitting up the data set downloaded from:"
[2] "https://www.ontario.ca/data/ontario-public-schools-enrolment"
[3] "under the Open Government License - Ontario:"
[4] "https://www.ontario.ca/page/open-government-licence-ontario"
> print(length(content1))
[1] 4
> print(nchar(content1))
[1] 74 60 44 59
```

R programming language for Data Science

5



## Text file

---

### ❑ Các phương thức

- Kích cỡ file: **file.size("Đường dẫn tập tin")**

#### ▪ Ví dụ:

```
> print(file.size("~/R/chuong11/Du_lieu/license.txt"))
[1] 242
```



## Text file

### □ Đọc file text với scan()

- Tạo ra các item từ các từ cách nhau bằng khoảng trắng

▪ Ví dụ:

```
> content1 <- scan("~/R/chuong11/Du_lieu/license.txt", "")  
Read 22 items  
> print(content1)  
[1] "These"  
[2] "data"  
[3] "files"  
[4] "are"  
[5] "derived"  
[6] "by"  
[7] "splitting"  
[8] "up"  
[9] "the"  
[10] "data"  
[11] "set"  
[12] "downloaded"  
[13] "from:"  
[14] "https://www.ontario.ca/data/ontario-public-schools-enrolment"  
[15] "under"  
[16] "the"  
[17] "Open"  
[18] "Government"  
[19] "License"  
[20] "_"  
[21] "Ontario:"  
[22] "https://www.ontario.ca/page/open-government-licence-ontario"  
R programming language for Data Science
```



## Text file

### □ Đọc file text có cấu trúc table

- Sử dụng **read.table**("Đường dẫn tập tin")
- Dữ liệu trả về là dạng table

▪ Ví dụ:

```
df <- read.table("~/R/chuong11/Du_lieu/Table0.txt")  
print(df)
```

	V1	V2	V3	V4	V5
1	Alex	25	177	57	F
2	Lilly	31	163	69	F
3	Mark	23	190	83	M
4	Oliver	52	179	75	M
5	Martha	76	163	70	F
6	Lucas	49	183	83	M
7	Caroline	26	164	53	F



## Text file

---

### □ Ghi file text chuỗi nội dung

- Sử dụng **write(nội\_dung, file= file\_name)**

- `file_name <- file(description = "Đường dẫn tập tin", open = "file_mode"[, encoding="UTF-8"])`

- Ví dụ:

```
> content2 <- "Hello and welcome"
> file_name <- file(description = "~/R/chuong11/Du_lieu/text.txt",open = "a", encoding="UTF-8")
> write(content2, file=file_name)
> content2 <- readLines("~/R/chuong11/Du_lieu/text.txt")
> print(content2)
[1] "Hello and welcome" "Hello and welcome"

> content2 <- matrix(c(1:9), nrow=3,ncol=3)
> write(content2, file "~/R/chuong11/Du_lieu/matrix.txt", ncolumns = 3, sep = " ")
> content2 <- readLines("~/R/chuong11/Du_lieu/matrix.txt")
> print(content2)
[1] "1 2 3" "4 5 6" "7 8 9"
```



## Text file

---

### □ Ghi file text dữ liệu có cấu trúc

- Sử dụng **write.table(nội\_dung, file= đường dẫn tập tin, sep="đ dấu phân cách",...)**

- Ví dụ:

```
write.table(df,
file = "~/R/chuong11/Du_lieu/Table0_new.txt",sep="\t")
```



## Nội dung

1. Text file
2. Csv file
3. Excel file



## Csv file



### □ Giới thiệu

- Định dạng CSV (Comma Separated Values) là một định dạng nhập/xuất dữ liệu phổ biến dành cho bảng tính và CSDL (spreadsheet & database).
- CSV file được sử dụng để lưu trữ một lượng lớn các biến hoặc dữ liệu. Đó là các bảng tính đơn giản – ví dụ như Excel – chỉ có nội dung được lưu trữ trong plaintext.



## Csv file

---

### □ Đọc file csv

- Sử dụng `read.csv("Đường dẫn tập tin")`,  
kết quả trả về có kiểu là data frame

▪ Ví dụ:

```
> # open and read csv file
> data <- read.csv("~/R/chuong11/Du_Lieu/emp.csv")
> print(data)
  id      name salary start_date      dept
1  1      Rick  623.30 2012-01-01        IT
2  2       Dan  515.20 2013-09-23 Operations
3  3 Michelle 611.00 2014-11-15        IT
4  4      Ryan  729.00 2014-05-11       HR
5 NA      Gary  843.25 2015-03-27 Finance
6  6      Nina  578.00 2013-05-21        IT
7  7     Simon  632.80 2013-07-30 Operations
8  8      Guru  722.50 2014-06-17 Finance
```

R programming language for Data Science

13



## Csv file

---

### □ Phân tích dữ liệu trong file

- `is.data.frame(file_content)`: kiểm tra kiểu  
của nội dung tập tin
- `ncol(file_content)`: trả về số cột
- `nrow(file_content)`: trả về số dòng

▪ Ví dụ:

```
> print(is.data.frame(data))
[1] TRUE
> print(ncol(data))
[1] 5
> print(nrow(data))
[1] 8
```

R programming language for Data Science

14



## Csv file

### □ Phân tích dữ liệu trong file

● Ví dụ:

```

id      name salary start_date      dept
1 1      Rick 623.30 2012-01-01      IT
2 2      Dan 515.20 2013-09-23 Operations
3 3 Michelle 611.00 2014-11-15      IT
4 4      Ryan 729.00 2014-05-11      HR
5 NA     Gary 843.25 2015-03-27 Finance
6 6      Nina 578.00 2013-05-21      IT
7 7      Simon 632.80 2013-07-30 Operations
8 8      Guru 722.50 2014-06-17 Finance

print(paste("Max salary:", max(data$salary)))
print(paste("Mean salary:", mean(data$salary)))
print("People have max salary:")
print(subset(data, salary == max(data$salary)))

[1] "Max salary: 843.25"
[1] "Mean salary: 656.88125"
[1] "People have max salary:"
id name salary start_date      dept
5 NA Gary 843.25 2015-03-27 Finance

```

R programming language for Data Science

15

## Csv file

```

> #Get all the people working in IT department
> retval <- subset( data, dept == "IT")
> print(retval)
  id      name salary start_date dept
1 1      Rick 623.3 2012-01-01  IT
3 3 Michelle 611.0 2014-11-15  IT
6 6      Nina 578.0 2013-05-21  IT
> #Get the persons in IT department whose salary is greater than 600
> info <- subset(data, salary > 600 & dept == "IT")
> print(info)
  id      name salary start_date dept
1 1      Rick 623.3 2012-01-01  IT
3 3 Michelle 611.0 2014-11-15  IT
> #Get the people who joined on or after 2014
> retval <- subset(data, as.Date(start_date) > as.Date("2014-01-01"))
> print(retval)
  id      name salary start_date      dept
3 3 Michelle 611.00 2014-11-15      IT
4 4      Ryan 729.00 2014-05-11      HR
5 NA     Gary 843.25 2015-03-27 Finance
8 8      Guru 722.50 2014-06-17 Finance

```

R programming language for Data Science

16

## Csv file

---

### □ Ghi csv file

- Sử dụng **write.csv(data\_frame\_name, “Đường dẫn tập tin”)**

- Ví dụ:

```
> #Get all the people working in IT department
> it_emps <- subset( data, dept == "IT")
> # write it_emps data into a new file.
> write.csv(it_emps, "~/R/chuong11/Du_lieu/it_emps.csv")
> newdata <- read.csv("~/R/chuong11/Du_lieu/it_emps.csv")
> print(newdata)
  id      name salary start_date dept
1 1      Rick  623.3 2012-01-01   IT
2 3 Michelle 611.0 2014-11-15   IT
3 6      Nina  578.0 2013-05-21   IT
```



## Csv file

---

- Ghi chú: cột X được tự động tạo ra khi ghi dữ liệu mới vào file => nếu không muốn ghi cột X thì dùng **row.names = FALSE**

- Ví dụ:

```
> # write it_emps data into a new file without X column
> write.csv(it_emps, "~/R/chuong11/Du_lieu/it_emps.csv", row.names = FALSE)
> newdata <- read.csv("~/R/chuong11/Du_lieu/it_emps.csv")
> print(newdata)
  id      name salary start_date dept
1 1      Rick  623.3 2012-01-01   IT
2 3 Michelle 611.0 2014-11-15   IT
3 6      Nina  578.0 2013-05-21   IT
```



## Nội dung

---

1. Text file
2. Csv file
3. Excel file



---

## Excel file

---

### Giới thiệu

- Excel nằm trong bộ công cụ văn phòng Microsoft Office. Phần mềm này giúp tạo ra các bảng tính, cùng các tính năng, công cụ hỗ trợ người dùng lưu trữ, tính toán dữ liệu với số lượng hàng triệu ô tính và nhiều sheet làm việc.



## Excel file

---

- Để làm việc với tập tin .xls hay .xlsx cần `install.packages("xlsx")`
- Và load “xlsx” package để làm việc

```
> # Load the library into R workspace.
> library("xlsx")
Loading required package: rJava
Loading required package: xlsxjars
```



## Excel file

---

- Đọc excel file
  - Sử dụng `read.xlsx("Đường dẫn tập tin", sheetIndex = thứ_tự_sheet)`
  - Ví dụ:

```
> # Read the first worksheet in the file emp_data.xlsx.
> data <- read.xlsx("~/R/chuong11/du_lieu/emp_data.xlsx", sheetIndex = 1)
> print(data)
   id    name salary start_date      dept
1  1     Rick  623.30 2012-01-01        IT
2  2      Dan  515.20 2013-09-23  Operations
3  3 Michelle  611.00 2014-11-15        IT
4  4     Ryan  729.00 2014-05-11       HR
5  5      Gary  843.25 2015-03-27  Finance
6  6      Nina  578.00 2013-05-21        IT
7  7    Simon  632.80 2013-07-30  Operations
8  8     Guru  722.50 2014-06-17  Finance
```



## Excel file

---

- Sử dụng **read.xlsx("Đường dẫn tập tin", sheetName = "tên\_sheet")**

- Ví dụ:

```
> # Read the first worksheet in the file emp_data.xlsx.
> data <- read.xlsx("~/R/chuong11/Du_lieu/emp_data.xlsx", sheetName = "city")
> print(data)
  name   city
1  Rick Seattle
2    Dan Tampa
3 Michelle Chicago
4  Ryan Seattle
5  Gary Houston
6  Nina Boston
7 Simon Mumbai
8  Guru Dallas
```



## Excel file

---

### □ Ghi file excel

- Sử dụng **write.xlsx(data\_frame\_name, file = "Đường dẫn tập tin", sheetName = "Tên\_sheet" [, col.names = TRUE, row.names = TRUE, append = FALSE hoặc TRUE, showNA = TRUE hoặc FALSE])**

- Ví dụ

```
> # Load the library into R workspace.
> library("xlsx")
> # write the first data set in a new workbook
> write.xlsx(retval, file = "../R/chuong11/Du_lieu/emp_data.xlsx",
+             sheetName = "Sheet1", append = FALSE)
> newdata <- read.xlsx("~/R/chuong11/Du_lieu/emp_data.xlsx", sheetName = "Sheet1")
> print(newdata)
  NA. id      name salary start_date dept
1   1     Rick  623.3 2012-01-01    IT
2   3  Michelle  611.0 2014-11-15    IT
3   6      Nina  578.0 2013-05-21    IT
```



# Chapter 9: Làm việc với tập tin text, csv, excel

## Exercise 1: Đọc tập tin và xử lý

- Cung cấp tập tin: SampleTextFile\_10kb.txt
- Đọc tập tin
- In ra nội dung của phần tử đầu tiên trong tập tin
- In số phần tử của tập tin
- Mỗi phần tử có bao nhiêu ký tự?

## Exercise 2: Đọc tập tin, xử lý, ghi nội dung vào tập tin

- Cung cấp tập tin: Table0.txt
- Đọc tập tin có cấu trúc dạng bảng. In nội dung tập tin

	V1	V2	V3	V4	V5
1		Alex	25	177	57
2		Lilly	31	163	69
3		Mark	23	190	83
4		Oliver	52	179	75
5		Martha	76	163	70
6		Lucas	49	183	83
7		Caroline	26	164	53

- Đặt tên cho các cột lần lượt là: 'Name', 'Age', 'Height', 'Weight', 'Sex'

	Name	Age	Height	Weight	Sex
1	Alex	25	177	57	F
2	Lilly	31	163	69	F
3	Mark	23	190	83	M
4	Oliver	52	179	75	M
5	Martha	76	163	70	F
6	Lucas	49	183	83	M
7	Caroline	26	164	53	F

- Đặt row.names là dữ liệu cột 'Name'
- Xóa bỏ cột 'Name' thừa
- In nội dung

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

- Ghi nội dung này vào tập tin Table0\_new.txt



## Exercise 3: Đọc, ghi tập tin txt

- Cung cấp tập tin: Table6.txt
- Đọc tập tin có cấu trúc dạng bảng, có láy header, bỏ đi các Strings as Factors, bỏ đi các comment => df. In head của df.
- Kiểm tra xem data có dữ liệu trùng không, nếu có thì xóa các dòng bị trùng
- In nội dung sau khi đã bỏ dòng trùng.

Name	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

- Đặt rownames là nội dung của cột Name, sau đó xóa bỏ cột Name. In kết quả.

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

- Ghi lại nội dung sau khi đã chuẩn hóa vào tập tin Table6\_new.txt

## Exercise 4: Đọc, xử lý, ghi tập tin csv

- Cung cấp tập tin: states3.csv
- Đọc tập tin, row.names là cột state, chú ý dấu phân cách thập phân và dấu tách => df. In head của df
- Từ df, tạo ra df1 có income > 4500 và population >10000

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
California	21198	5114	1.1	71.71	10.3	62.6	20	156361
Illinois	11197	5107	0.9	70.14	10.3	52.6	127	55748
New York	18076	4903	1.4	70.55	10.9	52.7	82	47831
Ohio	10735	4561	0.8	70.82	7.4	53.2	124	40975

- Ghi bộ dữ liệu này vào tập tin states\_income.csv
- Đọc và hiển thị lại dữ liệu vừa ghi vào tập tin

## Exercise 5: Đọc tập tin csv, xử lý, lưu vào tập tinxlsx



- Cung cấp tập tin: medals.csv
- Đọc nội dung => data. In head của data

Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
1924	Chamonix	Skating	Figure skating	AUT	individual		M Silver
1924	Chamonix	Skating	Figure skating	AUT	individual		W Gold
1924	Chamonix	Skating	Figure skating	AUT	pairs		X Gold
1924	Chamonix	Bobsleigh	Bobsleigh	BEL	four-man		M Bronze
1924	Chamonix	Ice Hockey	Ice Hockey	CAN	ice hockey		M Gold
1924	Chamonix	Biathlon	Biathlon	FIN	military patrol		M Silver

- Cho biết kiểu dữ liệu của data.
- Cho biết số dòng, số cột và tên các cột của data
- Tạo một bộ dữ liệu data\_1932\_skiing với Year = 1932 và Sport = "Skiing"
- Ghi df\_sub vào file excel medals\_new.xlsx, tên sheet là 1932\_skiing, cho phép ghi tiếp.

## Exercise 6: Đọc và xử lý tập tin xlsx

- Đọc nội dung tập tin đã tạo ở bài trên (medals\_new.xlsx) => data. In head của data
- Hiệu chỉnh lại nội dung bằng cách bỏ cột NA
- Cho biết có bao nhiêu huy chương vàng trong danh sách này
- Chi tiết về các huy chương vàng này

Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
2	1932	Lake Placid	Skiing	Cross Country S	FIN	50km	M Gold
6	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual	M Gold
9	1932	Lake Placid	Skiing	Ski Jumping	NOR	K90 individual (70m)	M Gold
11	1932	Lake Placid	Skiing	Cross Country S	SWE	18km	M Gold

## Gợi ý

## Exercise 1: Đọc tập tin và xử lý

```
In [2]: #SampleTextFile_10kb.txt
content <- readLines("Du_lieu/SampleTextFile_10kb.txt")
print("**** First element in content:")
print(content[1])
print(paste("**** Number of elements:", length(content)))
print("**** Number of characters in each element:")

[1] "**** First element in content:"
[1] "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus condimentum sagittis lacus, laoreet luctus ligula laoreet ut. Vestibulum ullamcorper accumsan velit vel vehicula. Proin tempor lacus arcu. Nunc at elit condimentum, semper nisi et, condimentum mi. In venenatis blandit nibh at sollicitudin. Vestibulum dapibus mauris at orci maximus pellentesque. Nullam id elementum ipsum. Suspendisse cursus lobortis viverra. Proin et erat at mauris tincidunt porttitor vitae ac dui."
[1] "**** Number of elements: 27"
[1] "**** Number of characters in each element:"
```

```
In [5]: i<-1
while(i<=length(content)){
  print(paste("Element", i, "has", nchar(content[i]), "chars"))
  i<- i+1
}

[1] "Element 1 has 482 chars"
[1] "Element 2 has 0 chars"
[1] "Element 3 has 931 chars"
[1] "Element 4 has 0 chars"
[1] "Element 5 has 878 chars"
[1] "Element 6 has 0 chars"
[1] "Element 7 has 601 chars"
[1] "Element 8 has 0 chars"
[1] "Element 9 has 645 chars"
[1] "Element 10 has 0 chars"
[1] "Element 11 has 730 chars"
[1] "Element 12 has 0 chars"
[1] "Element 13 has 437 chars"
[1] "Element 14 has 0 chars"
[1] "Element 15 has 434 chars"
[1] "Element 16 has 0 chars"
[1] "Element 17 has 944 chars"
[1] "Element 18 has 0 chars"
[1] "Element 19 has 629 chars"
[1] "Element 20 has 0 chars"
[1] "Element 21 has 772 chars"
[1] "Element 22 has 0 chars"
[1] "Element 23 has 615 chars"
[1] "Element 24 has 0 chars"
[1] "Element 25 has 449 chars"
[1] "Element 26 has 0 chars"
[1] "Element 27 has 909 chars"
```

## Exercise 2: Đọc tập tin, xử lý, ghi nội dung vào tập tin

```
In [8]: df <- read.table("Du_lieu/Table0.txt")
print("Content of file:")
print(df)
```

```
[1] "Content of file:"
      V1 V2 V3 V4 V5
1    Alex 25 177 57 F
2   Lilly 31 163 69 F
3   Mark 23 190 83 M
4 Oliver 52 179 75 M
5 Martha 76 163 70 F
6 Lucas 49 183 83 M
7 Caroline 26 164 53 F
```

```
In [9]: #dat ten cho cac cot
names(df) <- c('Name', 'Age', 'Height', 'Weight', 'Sex')
print(df)
```

	Name	Age	Height	Weight	Sex
1	Alex	25	177	57	F
2	Lilly	31	163	69	F
3	Mark	23	190	83	M
4	Oliver	52	179	75	M
5	Martha	76	163	70	F
6	Lucas	49	183	83	M
7	Caroline	26	164	53	F

```
In [10]: #dat rowname La du Lieu cot Name
row.names(df)<-df$Name
print(df)
```

	Name	Age	Height	Weight	Sex
Alex	Alex	25	177	57	F
Lilly	Lilly	31	163	69	F
Mark	Mark	23	190	83	M
Oliver	Oliver	52	179	75	M
Martha	Martha	76	163	70	F
Lucas	Lucas	49	183	83	M
Caroline	Caroline	26	164	53	F

```
In [11]: #sau do bo di cot name thua
df$name <- NULL
print("Content of data frame:")
print(df)
```

```
[1] "Content of data frame:"
      Age Height Weight Sex
Alex     25     177     57 F
Lilly    31     163     69 F
Mark     23     190     83 M
Oliver   52     179     75 M
Martha   76     163     70 F
Lucas    49     183     83 M
Caroline 26     164     53 F
```

```
In [12]: # ghi noi dung tap tin
    print("Write to file...")
    write.table(df,file="Du_lieu/Table0_new.txt",sep="\t")
    print("Write completely!")

[1] "Write to file..."
[1] "Write completely!"
```

```
In [28]: df_new <- read.table("Du_lieu/Table0_new.txt")
    print("Read saved file:")
    print(df_new)

[1] "Read saved file:"
      Age Height Weight Sex
Alex     25     177     57   F
Lilly    31     163     69   F
Mark     23     190     83   M
Oliver   52     179     75   M
Martha   76     163     70   F
Lucas    49     183     83   M
Caroline 26     164     53   F
```

## Exercise 3: Đọc, ghi tập tin txt

```
In [29]: # doc du lieu
    # co lay header
    # bo di cac strings As Factors
    # bo di cac comment
    df <- read.table("Du_lieu/Table6.txt",
                      skip = 1,
                      header = TRUE,
                      flush = TRUE,
                      comment.char = "@",
                      stringsAsFactors = FALSE)
    print("Content of file after reading:")
    print(head(df))

[1] "Content of file after reading:"
      Name Age Height Weight Sex
1   Alex  25     177     57   F
2   Lilly 31     163     69   F
3   Mark  23     190     83   M
4 Oliver 52     179     75   M
5 Martha 76     163     70   F
6 Lucas  49     183     83   M
```

```
In [30]: print("Before dropping: ")
    print(paste("Rows:", nrow(df)))

[1] "Before dropping: "
[1] "Rows: 105"
```

```
In [31]: library(tidyverse)
```

```
In [32]: df = df %>% unique()
print("After dropping: ")
print(paste("Rows:", nrow(df)))

[1] "After dropping: "
[1] "Rows: 7"
```

```
In [33]: df
```

Name	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

```
In [34]: rownames(df) <- df$name
df$name <- NULL
df
```

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

```
In [35]: # ghi noi dung tap tin
print("Write to file...")
write.table(df,file="Du_lieu/Table6_new.txt",sep="\t")
print("Write completely!")

[1] "Write to file..."
[1] "Write completely!"
```

```
In [36]: df_new <- read.table("Du_lieu/Table6_new.txt")
print("Read saved file:")
print(df_new)
```

```
[1] "Read saved file:"
  Age Height Weight Sex
Alex    25     177     57   F
Lilly   31     163     69   F
Mark    23     190     83   M
Oliver  52     179     75   M
Martha  76     163     70   F
Lucas   49     183     83   M
Caroline 26     164     53   F
```

## Exercise 4: Đọc, xử lý, ghi tập tin csv

```
In [38]: df <- read.csv("Du_lieu/states2.csv",
                      row.names = 1,
                      sep = ";",
                      dec = ",")
print(head(df))
```

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
Alabama	3615	3624	2.1	69.05	15.1	41.3	20	50708
Alaska	365	6315	1.5	69.31	11.3	66.7	152	566432
Arizona	2212	4530	1.8	70.55	7.8	58.1	15	113417
Arkansas	2110	3378	1.9	70.66	10.1	39.9	65	51945
California	21198	5114	1.1	71.71	10.3	62.6	20	156361
Colorado	2541	4884	0.7	72.06	6.8	63.9	166	103766

```
In [39]: # tao ra bo du Lieu ma income >4500 va population >10000
df1 <- subset(df, df$Income > 4500 & df$Population >10000)
print("Data set of income >4500 and population >10000:")
print(df1)
```

```
[1] "Data set of income >4500 and population >10000:"
```

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
California	21198	5114	1.1	71.71	10.3	62.6	20	156361
Illinois	11197	5107	0.9	70.14	10.3	52.6	127	55748
New York	18076	4903	1.4	70.55	10.9	52.7	82	47831
Ohio	10735	4561	0.8	70.82	7.4	53.2	124	40975

```
In [40]: # luu bo du Lieu nay vao tap tin states_income.csv
write.csv(df1, file = "Du_lieu/states_income.csv")
```

```
In [41]: #doc du lieu trong file states_income.csv
dfa <- read.csv("Du_lieu/states_income.csv", row.names = 1)
print("Data in states_income.csv ")
print(dfa)

[1] "Data in states_income.csv "
  Population Income Illiteracy Life.Exp Murder HS.Grad Frost Area
California     21198    5114      1.1    71.71   10.3    62.6    20 156361
Illinois       11197    5107      0.9    70.14   10.3    52.6    127 55748
New York       18076    4903      1.4    70.55   10.9    52.7     82 47831
Ohio            10735    4561      0.8    70.82    7.4    53.2    124 40975
```

```
In [42]: # solution 2
library(tidyverse)
```

```
In [43]: df11 <- df %>% filter(Income > 4500, df$Population > 10000)
df11
```

Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
21198	5114	1.1	71.71	10.3	62.6	20	156361
11197	5107	0.9	70.14	10.3	52.6	127	55748
18076	4903	1.4	70.55	10.9	52.7	82	47831
10735	4561	0.8	70.82	7.4	53.2	124	40975

## Exercise 5: Đọc tập tin csv, xử lý, lưu vào tập tin xlsx

```
In [44]: # Load the library into R workspace.
```

```
library("xlsx")
```

```
Loading required package: rJava
```

```
Loading required package: xlsxjars
```

```
In [47]: #doc tap tin school.csv
data <- read.csv("Du_lieu/medals.csv")
head(data)
```

Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
1924	Chamonix	Skating	Figure skating	AUT	individual		M Silver
1924	Chamonix	Skating	Figure skating	AUT	individual		W Gold
1924	Chamonix	Skating	Figure skating	AUT	pairs		X Gold
1924	Chamonix	Bobsleigh	Bobsleigh	BEL	four-man		M Bronze
1924	Chamonix	Ice Hockey	Ice Hockey	CAN	ice hockey		M Gold
1924	Chamonix	Biathlon	Biathlon	FIN	military patrol		M Silver



```
In [48]: print(paste("Data type: ",class(data)))
print(paste("Number of rows:", ncol(data)))
print(paste("Number of cols:", nrow(data)))
print(paste("Name of columns", toString(names(data)))))

[1] "Data type: data.frame"
[1] "Number of rows: 8"
[1] "Number of cols: 2314"
[1] "Name of columns Year, City, Sport, Discipline, NOC, Event, Event.gender, Medal"
```

```
In [49]: # tao bo du Lieu co year = 1932, Sport = 'Skiing'
data_1932_skiing <- subset(data, data$Year == 1932 & data$Sport=='Skiing')
print("Data set of year = 1932 and Sport = Skiing:")
data_1932_skiing

[1] "Data set of year = 1932 and Sport = Skiing:"
```

	Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
100	1932	Lake Placid	Skiing	Cross Country S	FIN	18km	M	Bronze
101	1932	Lake Placid	Skiing	Cross Country S	FIN	50km	M	Gold
102	1932	Lake Placid	Skiing	Cross Country S	FIN	50km	M	Silver
110	1932	Lake Placid	Skiing	Cross Country S	NOR	50km	M	Bronze
111	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual	M	Bronze
112	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual	M	Gold
113	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual	M	Silver
114	1932	Lake Placid	Skiing	Ski Jumping	NOR	K90 individual (70m)	M	Bronze
115	1932	Lake Placid	Skiing	Ski Jumping	NOR	K90 individual (70m)	M	Gold
116	1932	Lake Placid	Skiing	Ski Jumping	NOR	K90 individual (70m)	M	Silver
119	1932	Lake Placid	Skiing	Cross Country S	SWE	18km	M	Gold
120	1932	Lake Placid	Skiing	Cross Country S	SWE	18km	M	Silver

```
In [50]: # ghi bo du lieu vao file excel medals_new.xlsx
print("Write to file...")
write.xlsx(data_1932_skiing, file = "Du_lieu/medals_new.xlsx",
           sheetName = "1932_skiing", append = TRUE)
print("Finish!")

[1] "Write to file..."
[1] "Finish!"
```

## Exercise 6: Đọc và xử lý tập tinxlsx



In [56]: # Read the first worksheet in the file medals\_1.xlsx  
 data\_1932\_skiing <- read.xlsx("Du\_lieu/medals\_new.xlsx", sheetName = "1932\_SKIING")  
 print("Content of file:")  
 head(data\_1932\_skiing)

[1] "Content of file:"

NA.	Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
100	1932	Lake Placid	Skiing	Cross Country S	FIN	18km		M Bronze
101	1932	Lake Placid	Skiing	Cross Country S	FIN	50km		M Gold
102	1932	Lake Placid	Skiing	Cross Country S	FIN	50km		M Silver
110	1932	Lake Placid	Skiing	Cross Country S	NOR	50km		M Bronze
111	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual		M Bronze
112	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual		M Gold

In [57]: #xoa bo cot NA.

```
data_1932_skiing$NA. <- NULL
print("Data set now")
print(head(data_1932_skiing))
```

[1] "Data set now"

	Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
1	1932	Lake Placid	Skiing	Cross Country S	FIN	18km		M Bronze
2	1932	Lake Placid	Skiing	Cross Country S	FIN	50km		M Gold
3	1932	Lake Placid	Skiing	Cross Country S	FIN	50km		M Silver
4	1932	Lake Placid	Skiing	Cross Country S	NOR	50km		M Bronze
5	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual		M Bronze
6	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual		M Gold

In [58]: #cho biet co bao nhieu huy chuong "Gold" trong du lieu nay

```
n <- sum(data_1932_skiing$Medal == "Gold")
print(paste("Number of Golden medals:", n))
```

[1] "Number of Golden medals: 4"

In [60]: #thong tin chi tiet

```
data_golden_medals <- subset(data_1932_skiing, data_1932_skiing$Medal == "Gold")
print("Details:")
data_golden_medals
```

[1] "Details:"

	Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
2	1932	Lake Placid	Skiing	Cross Country S	FIN	50km		M Gold
6	1932	Lake Placid	Skiing	Nordic Combined	NOR	individual		M Gold
9	1932	Lake Placid	Skiing	Ski Jumping	NOR	K90 individual (70m)		M Gold
11	1932	Lake Placid	Skiing	Cross Country S	SWE	18km		M Gold