

R 語言簡介

Sung-Chien Lin

2018 年 8 月 25 日

課程簡介

課程簡介

- 本次課程為 R 語言的基礎介紹
- 主要目的在熟悉 R 語言的開發環境與基本用法
- 重點在 R 語言的基本與特殊的資料型態

學習目標

- 能夠安裝 R 語言的開發環境，包括核心程式以及整合開發環境(Integrated Development Environment, IDE)
- 能夠運用 IDE 的 Console 撰寫 R 語言的指令
- 能夠區別 R 語言的各種基本資料型態：整數(integer)、數值(numeric)、文字(character)、邏輯(logic)
- 能夠區別 R 語言的各種進階資料型態：向量(vector)、矩陣(matrix)、列表(list)、資料框(data frame)

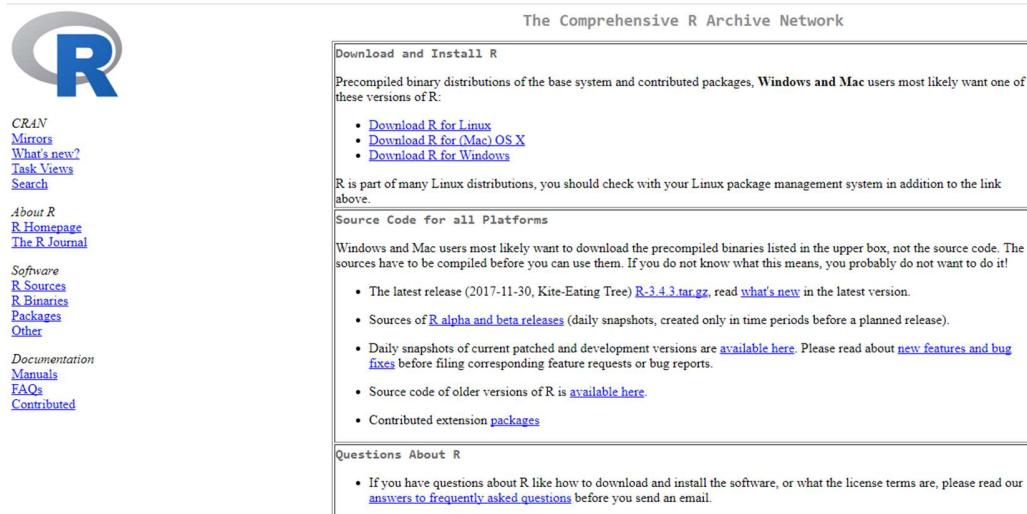
R 語言基礎

R 語言簡介

- 自由的軟體程式語言
- 用於統計分析、資料探勘、資料視覺化
 - 內建多種統計及資料分析功能
 - 可透過套件增強功能

R 的資源

- CRAN：下載 R 的執行檔、原始碼和說明文件，也收錄了各種套件
- 請連線到 <https://cran.r-project.org/>
- 點選 Download R for (Mac) OS X 或 Download for Windows 連結，下載 R 的核心程式，進行安裝
- **注意目前的版本**(R 常常會有新的版本釋出，若有需要可下載更新)



The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, Windows and Mac users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

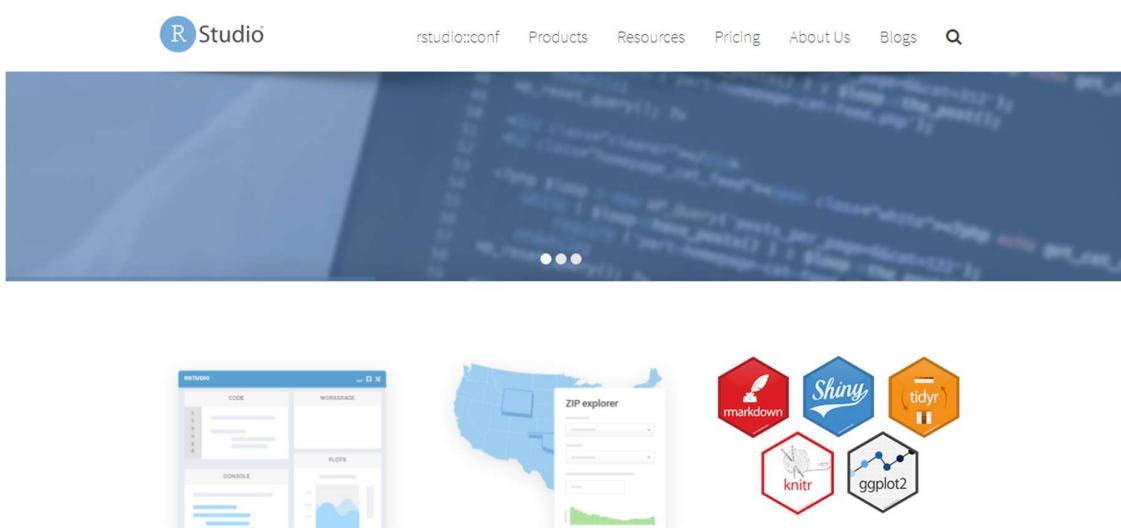
- The latest release (2017-11-30, Kite-Eating Tree) [R-3.4.3.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

R 的編寫介面

- R 本身的介面不容易用於開發程式
- RStudio : 輔助 R 語言程式開發的 IDE 應用軟體
- 請連線到 <https://www.rstudio.com/>
- 點選 Products 下的 RStudio 下載，安裝 RStudio

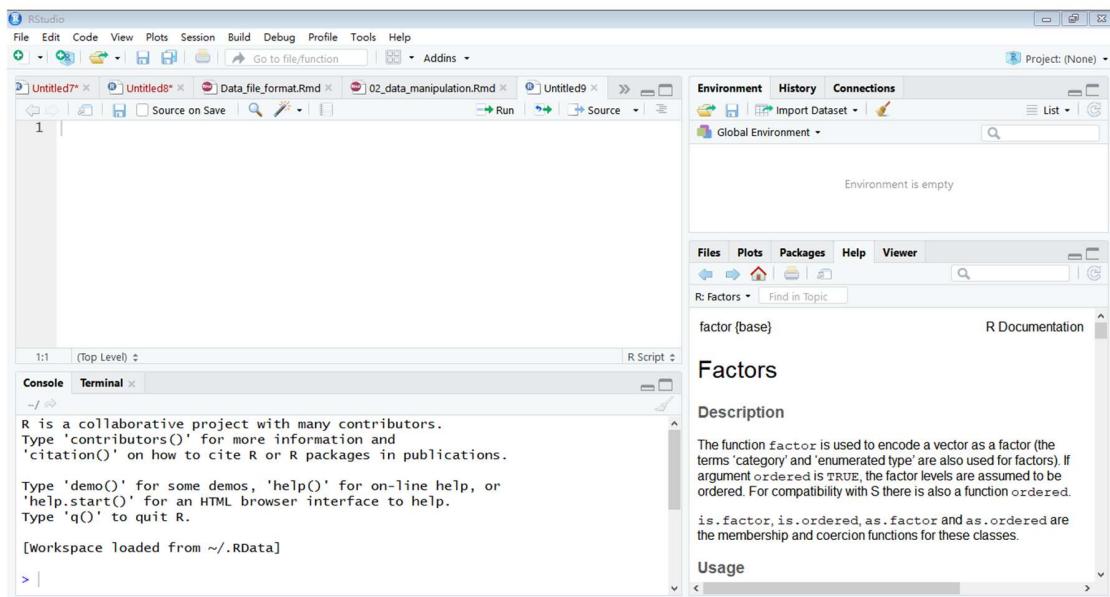


練習

- 安裝 R 核心程式與 RStudio

RStudio 介面

- 請開啟 RStudio



Rstudio 的儀表板

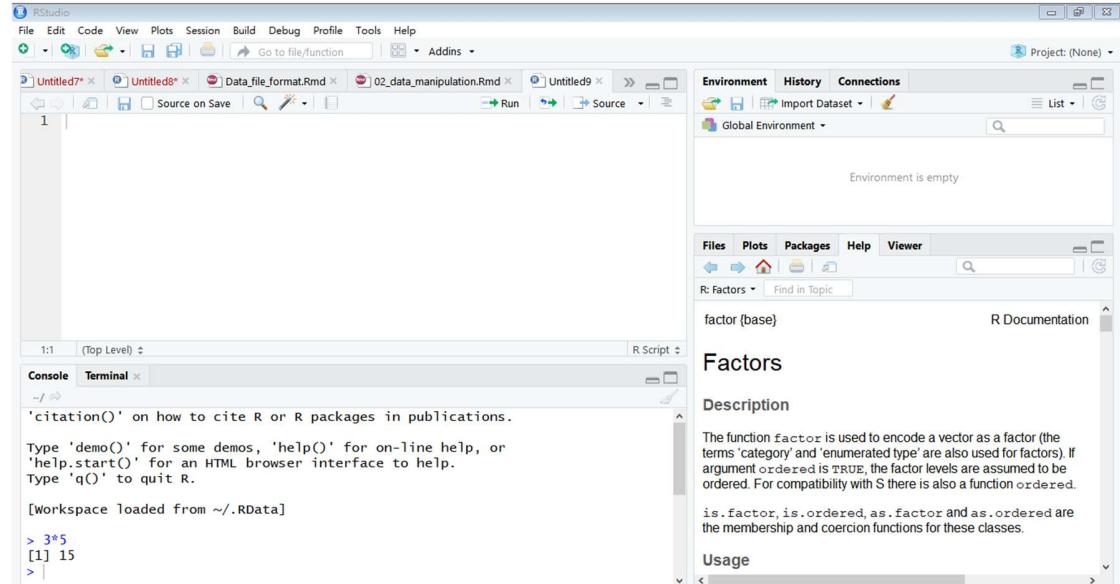
- Console：執行敘述，呈現敘述執行的結果(左下)
- Source：程式開發(左上)
- Environment：目前使用的變數與自行定義的函數(右上)
- Help：查詢函數(右下)
- Plots：繪圖結果(右下)

R 語言基礎

運算

- R 語言是一種直譯式語言(interpreted language)，在 Console 處執行運算，直接便可以看到結果
- 例如：請在 RStudio 的 Console 處輸入

```
3*5
```



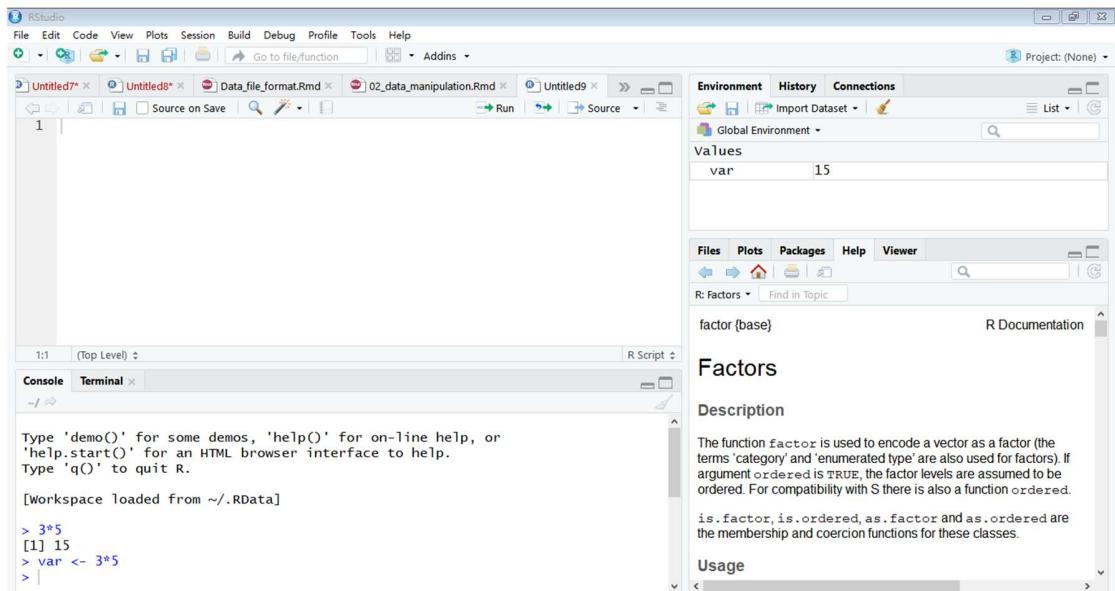
練習

- 試著在 Console 處執行其他運算，看看結果

指定敘述

- 可將運算結果儲存在某一個變數內
- 例如：在 Console 處輸入下面的指定敘述

```
var <- 3*5
```

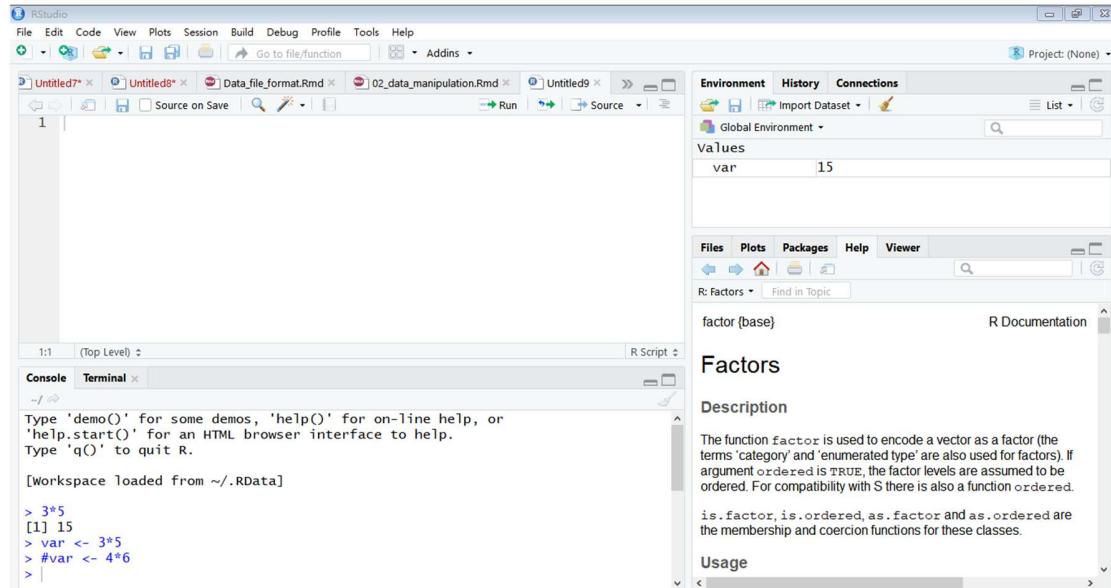


- 注意：`<-`的意思是將右邊的運算結果指定給左邊的變數儲存
- 在右上的 Environment，查看變數 var 儲存的數值

註解

- '#'後的文字被忽略不考慮
- 例如：在 Console 處再輸入下面的敘述

```
#var <- 4*6
```



- 在右上的 Environment，查看變數 var 儲存的數值
- 新的敘述並沒有覆蓋舊的指定敘述，所以 var 的值仍為是 15

查詢與設定工作目錄

- 通常開始一個任務時，會將相關的程式和資料儲存在同一個資料夾，稱為**工作目錄(working directory)**
- 使用工作目錄進行任務，可以減少不少錯誤發生，提高工作效率
- 查看目前的工作目錄

```
getwd()
```

- 在 Console 上，設定工作目錄

```
setwd("C:/工作路徑/rCourse/任務 X")
```

- 事實上，setwd 和 getwd 都是函數(function)，在 R 語言裡函數的使用相當重要

練習

- 試著在檔案總管中建立一個新的資料夾，並且使用 setwd() 將該資料夾設定為工作目錄

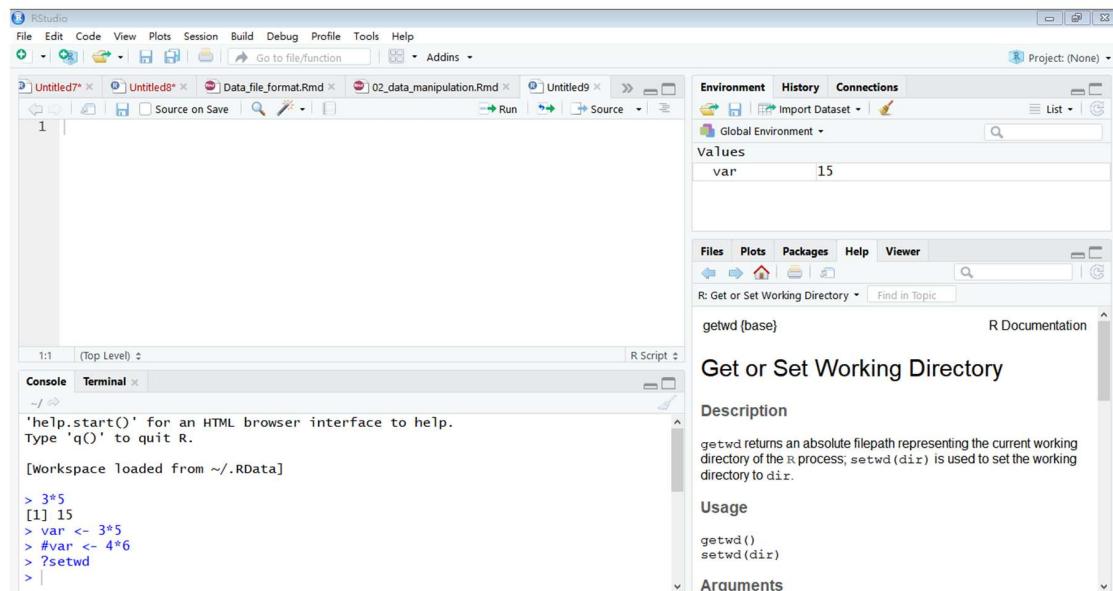
Help

- 在 Console 上，查看某一個函數的用法

```
help(setwd)
```

- 另一種常用的寫法

```
?setwd
```



- 右下的 Help 出現有關這個函數的解說

R 基本資料型態

R 基本資料型態

- integer
- numeric
- character
- logic

integer & numeric

- 數值資料
 - integer : 整數
 - numeric : 數值

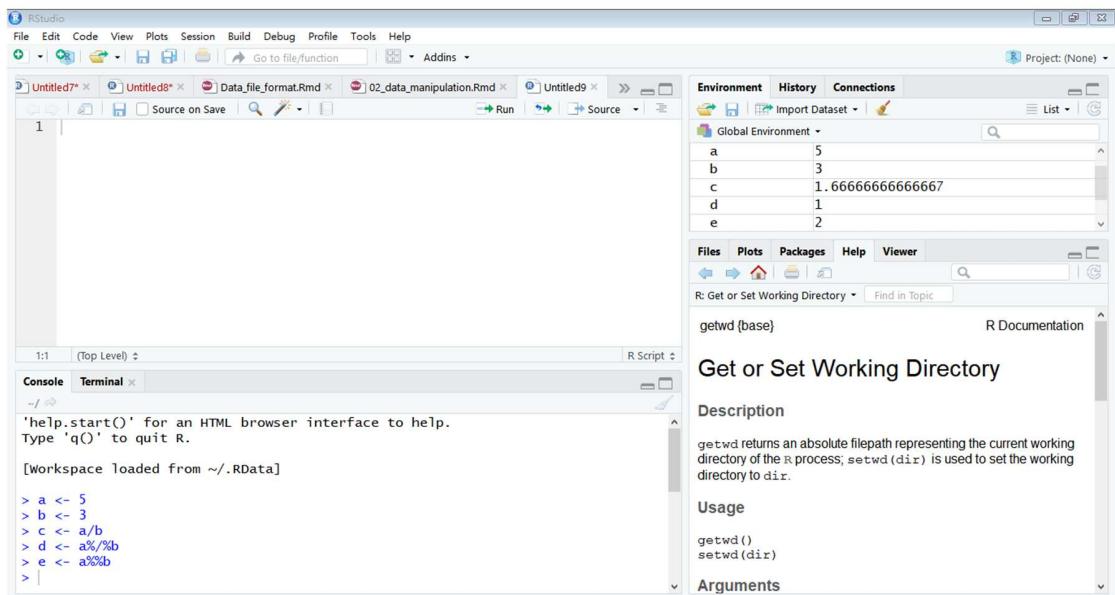
數值資料運算

- + 加
- - 減
- * 乘
- / 除
- ^ 次方
- %% 整數除法的餘數
- %/% 整數除法的商

練習

- 在 Console 下輸入下面的敘述，分別查看 c, d, e，說明為何 c, d, e 的數值為何不同？(可參考下頁的執行結果)

```
a <- 5  
b <- 3  
c <- a/b  
d <- a%/%b  
e <- a%%b
```



數值函數

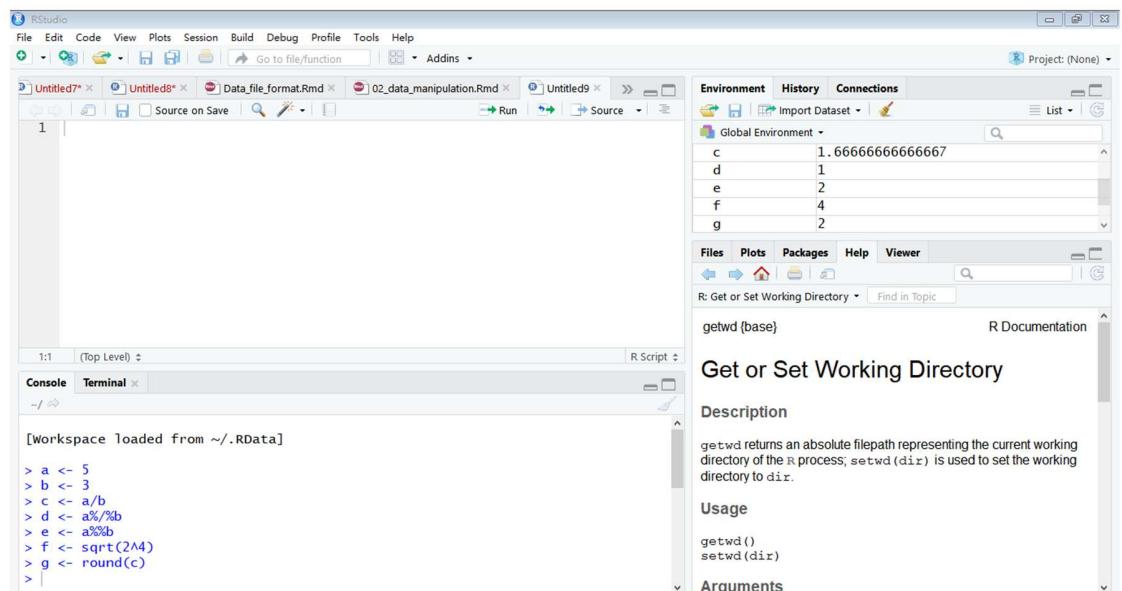
- `abs()` 絶對值
- `sqrt()` 平方根
- `round()` 四捨五入
- 注意：函數是學習程式語言裡最重要的一個環節，會使用函數的話，可以大大節省開發的時間和精力。很多函數都是已經開發好的，但有必要的話，也可以自行開發。已經開發的函數又可以分為兩類，一類是系統本身提供的函數，或是由其他人員寫好的套件(package)。目前我們使用的函數都是系統本身提供的函數，日後會大量使用套件，也會自行開發函數。

練習

- 在 Console 下輸入下面的敘述，分別查看 `f, g`

```
f <- sqrt(2^4)
```

```
g <- round(c)
```



- 注意：函數括號裡的資料稱為參數(parameter)。輸入參數，經由函數運算，得到結果。

character

- character 資料表示
- 在 Console 上輸入下面指定敘述，並且查看 strR 和 strVar

```
strR <- "R"
```

```
strVar <- "Language"
```

character 資料的函數

- nchar() 字串內有多少字
- paste() 字串連接
- substr() 取出字串的一部份

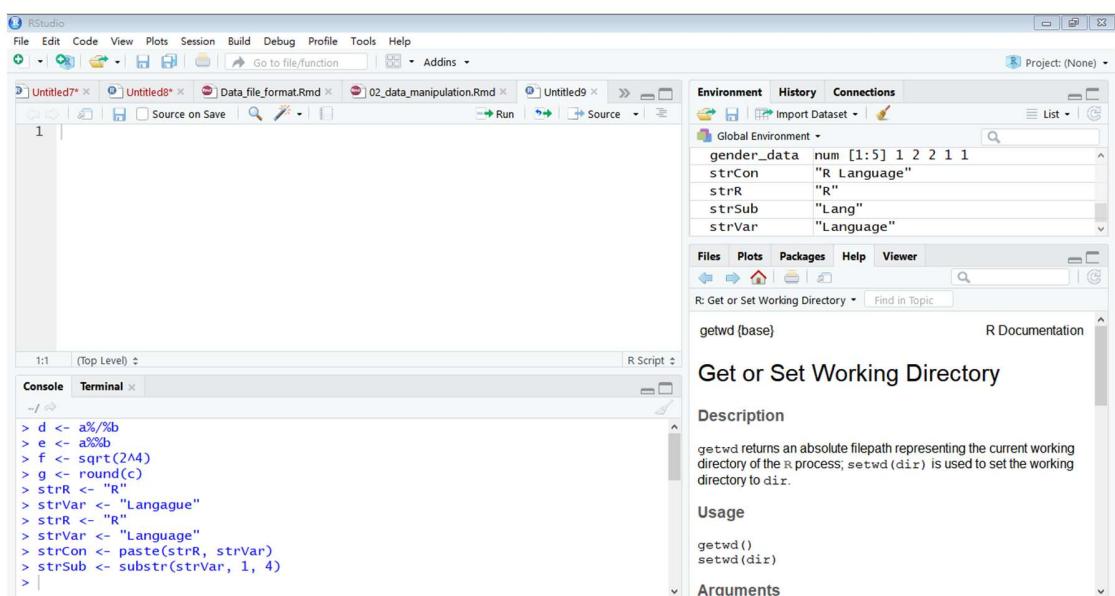
練習

- 在 Console 上輸入下面敘述，並且查看運算結果

```
nchar(strVar)
```

```
strCon <- paste(strR, strVar)
```

```
strSub <- substr(strVar, 1, 4)
```



logic

- TRUE(真)與 FALSE(偽)

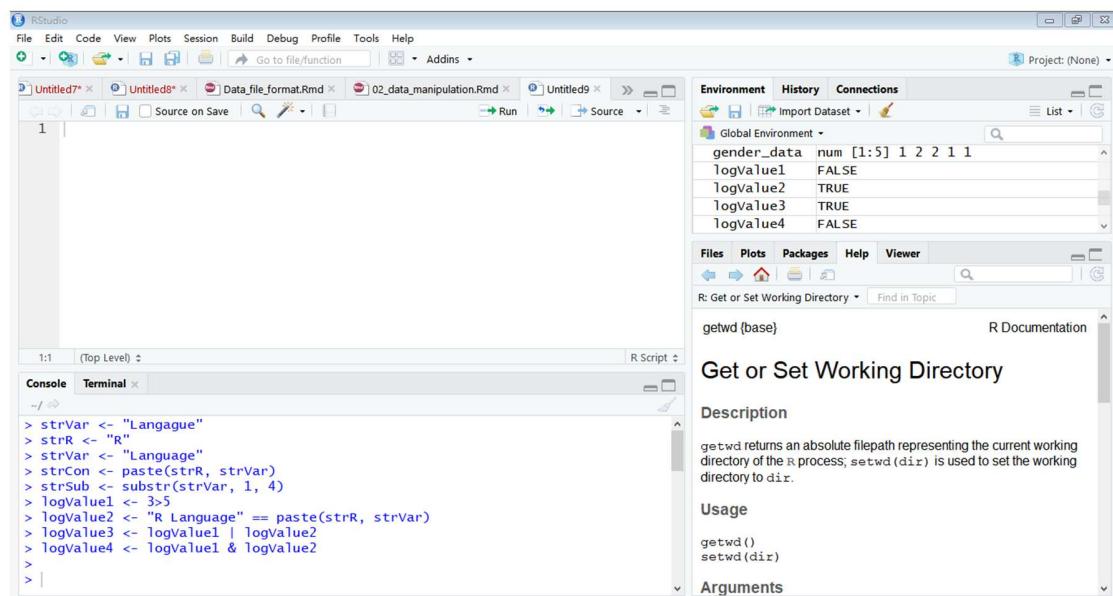
邏輯運算

- | 或：只要有一個 TRUE，便是 TRUE
- & 且：只要有一個 FALSE，便是 FALSE
- ! 否：TRUE 的結果為 FALSE，FALSE 的結果為 TRUE

練習

- 在 Console 上輸入下面敘述，並且查看運算結果

```
logValue1 <- 3>5  
  
logValue2 <- "R Language" == paste(strR, strVar)  
  
logValue3 <- logValue1 | logValue2  
  
logValue4 <- logValue1 & logValue2
```



R 特殊資料型態

R 特殊資料型態

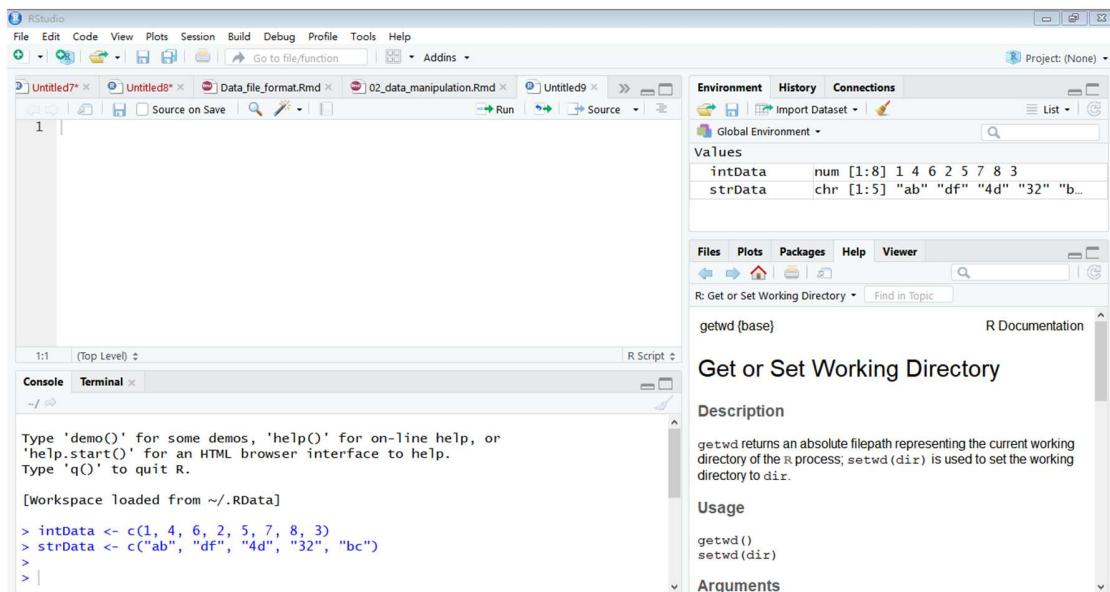
- vector
- matrix
- list
- data frame

vector

- 一序列相同型態的資料
- 在 Console 上輸入下面敘述，並且查看運算結果

```
intData <- c(1, 4, 6, 2, 5, 7, 8, 3)  
  
strData <- c("ab", "df", "4d", "32", "bc")
```

- 在右側的 Environment 上，觀察 vector 可以發現：intData 為 num[1:8]，而 strData 為 chr[1:5]
 - num, chr 分別表示 numeric 和 character 資料型態
 - [1:6], [1:5] vector 的範圍



vector

- 元素 : vector 上的資料，可以按照其位置取得，甚至替換
- 在 Console 上輸入下面敘述，並且查看運算結果

```
intData[4]
```

```
intData[2:5]
```

```
intData[c(1,3,5)]
```

- intData[4]表示 intData 的第 2 個元素
- intData[2:5]表示 intData 的第 2 到第 5 個元素
- intData[c(1,3,5)]表示 intData 的第 1、3、5 個元素

vector 運算

- 數值運算：針對 vector 上的每一個資料，逐一進行運算
- 在 Console 上輸入下面敘述，並且查看運算結果

```
intData * 5
```

```
intData + c(1, 2, 3, 4, 5, 6, 7, 8)
```

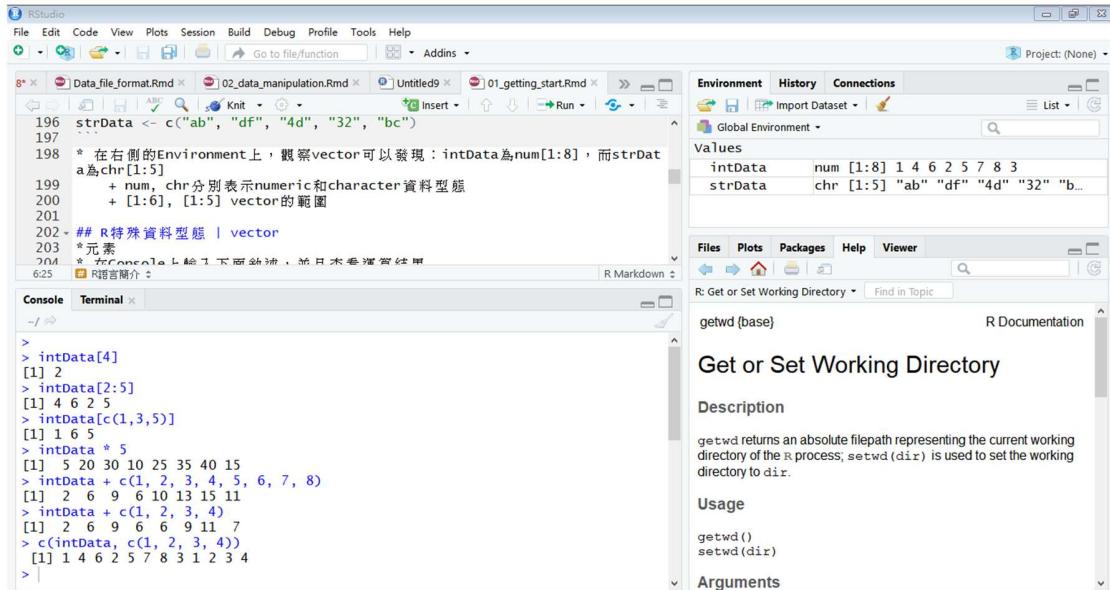
```
intData + c(1, 2, 3, 4)
```

- 注意：當兩個 vector 進行運算時，較短的 vector 會自動延長，以便兩個 vector 上面的元素一一對應

合併 vector

- `c()`：合併兩個或以上的 vectors
- 在 Console 上輸入下面敘述，並且查看運算結果

```
c(intData, c(1, 2, 3, 4))
```



vector 的長度

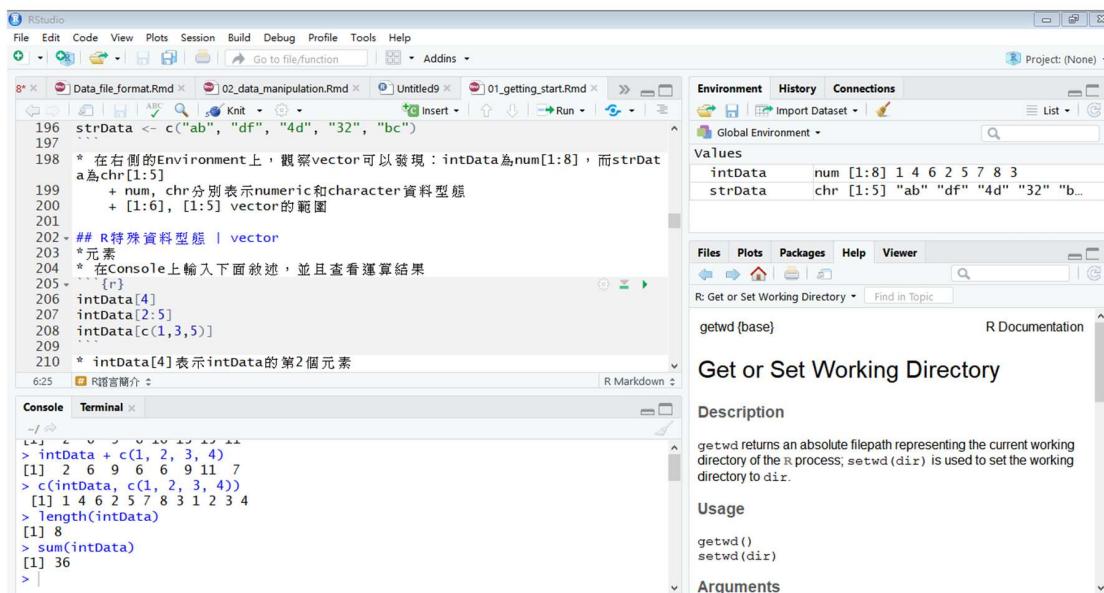
- `length()` : vector 的長度，也就是 vector 上的元素個數
- 在 Console 上輸入下面敘述，並且查看運算結果

```
length(intData)
```

vector 上的元素總和

- `sum()` : 計算 vector 上的元素總和
- 在 Console 上輸入下面敘述，並且查看運算結果

```
sum(intData)
```



排序 vector 上的元素

- `sort()`：對 vector 上的元素排序
- 在 Console 上輸入下面敘述，並且查看運算結果

```
sort(intData)
```

```
sort(intData, decreasing=TRUE)
```

- 注意：`sort()`函數中預設的 `decreasing` 為 `FALSE`，將 vector 裡的元素由小到大排序；將 `decreasing` 設為 `TRUE`，將 vector 裡的元素由大到小排列

The screenshot shows the RStudio interface. In the top-left, there are several Rmd files open: Data_file_format.Rmd, 02_data_manipulation.Rmd, Untitled9, and 01_getting_start.Rmd. The main workspace shows two vectors defined:

```
intData     num [1:8] 1 4 6 2 5 7 8 3
strData    chr [1:5] "ab" "df" "4d" "32" "b..."
```

In the bottom-left, the Console tab is active, displaying the following R session:

```
> length(intData)
[1] 8
> sum(intData)
[1] 36
> sort(intData)
[1] 1 2 3 4 5 6 7 8
> sort(intData, decreasing=TRUE)
[1] 8 7 6 5 4 3 2 1
```

The right side of the interface includes the Environment pane showing variable definitions, and a Help pane for the `getwd()` function.

產生等差序列的 vector

- `seq(from, to, by)`：需要三個參數。從開始的數值 `from`，在不超過結尾數值 `by` 的情形下，每次增加一個數值 `by`
- 在 Console 上輸入下面敘述，並且查看運算結果

```
seq(1, 8)
```

```
seq(1, 8, 2)
```

```
seq(1, 8, 1.5)
```

- `seq(1, 8)`: `from=1, to=8`
- `seq(1, 8, 2)`: `from=1, to=8, by=2`
- `seq(1, 8, 1.5)`: `from=1, to=8, by=1.5`

產生等差序列的 vector

- 如果差為 1 時，可以用冒號運算式
- 在 Console 上輸入下面敘述，並且查看運算結果

```
1:8
```

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** Displays R code related to data manipulation, including comments about vectors and their ranges.
- Console:** Shows the execution of `seq(1, 8)`, `seq(1, 8, 2)`, and `seq(1, 8, 1.5)`, along with the output vectors.
- Environment:** Shows the global environment with objects `intData` and `strData`.
- Help:** A detailed help page for the `getwd` function is displayed, including its description, usage, and arguments.

練習

- 產生以下的數列
 - 1 到 10 間所有奇數
 - 1 到 10 間所有偶數
 - 1 到 10 間所有 3 的倍數
 - 2 的 1 到 10 乘方

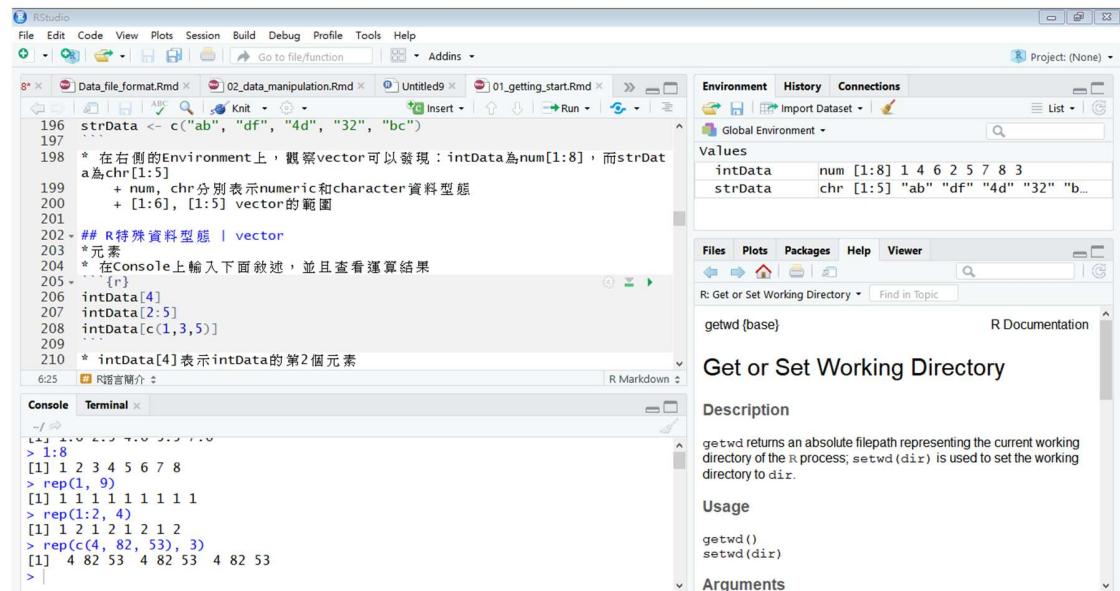
產生重複序列的 vector

- `rep(x, times)`：將 `x` 重複 `times` 次
- 在 Console 上輸入下面敘述，並且查看運算結果

```
rep(1, 9)
```

```
rep(1:2, 4)
```

```
rep(c(4, 82, 53), 3)
```



練習

- 運用 `length()` 計算上面序列的長度

matrix

- 2 維的 vector
- 在 Console 上輸入下面敘述

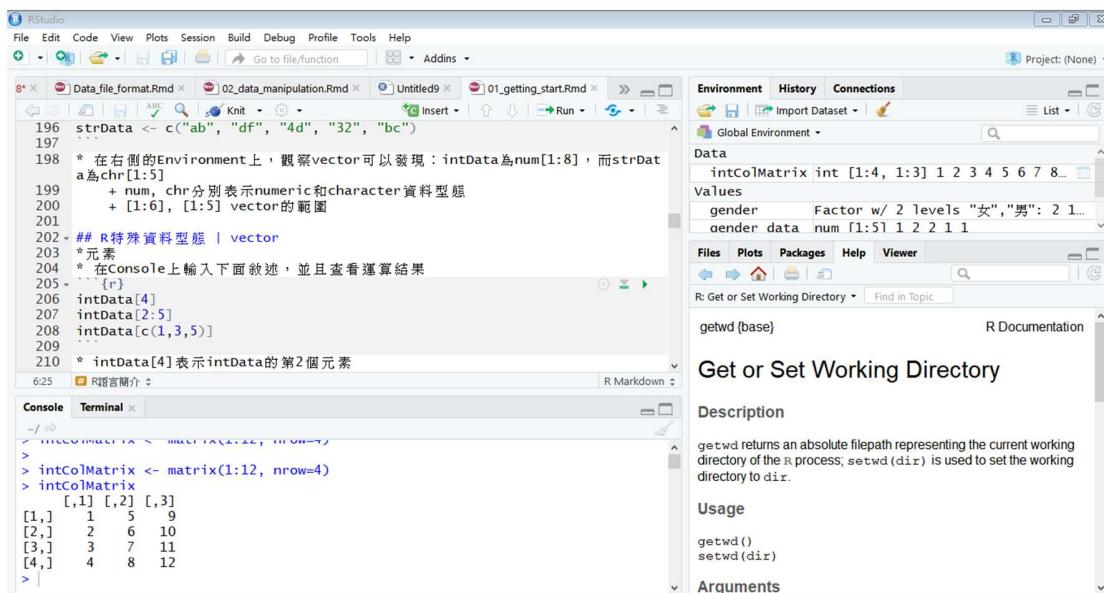
```
intColMatrix <- matrix(1:12, nrow=4)

intColMatrix

intRowMatrix <- matrix(1:12, nrow=4, byrow=TRUE)

intRowMatrix
```

- 在 matrix 裡，橫列稱為 row，直行稱為 col
- matrix(data, nrow, ncol, byrow)：將 data 上的資料表示成 nrow 列或 ncol 行的 matrix
 - byrow=FALSE(預設值)，直行優先排列
 - byrow=TRUE，橫列優先排列



The screenshot shows the RStudio interface. In the top-left, there are several Rmd files: Data_file_format.Rmd, 02_data_manipulation.Rmd, Untitled9.Rmd, and 01_getting_start.Rmd. The 02_data_manipulation.Rmd file is open, showing R code. The code includes comments explaining vector types and indexing. A matrix `intRowMatrix` is defined and printed to the console, showing rows 1 through 4 of a 12x4 matrix.

```

196 strData <- c("ab", "df", "4d", "32", "bc")
197 #> [1] "ab" "df" "4d" "32" "bc"
198 * 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData
  a為chr[1:5]
199   + num, chr分別表示numeric和character資料型態
200   + [1:6], [1:5] vector的範圍
201
202 ## R特殊資料型態 | vector
203 * 元素
204 * 在Console上輸入下面敘述，並且查看運算結果
205 * {r}
206 intData[4]
207 intData[2:5]
208 intData[c(1,3,5)]
209
210 * intData[4]表示intData的第2個元素

```

```

> intRowMatrix <- matrix(1:12, nrow=4, byrow=TRUE)
> intRowMatrix
     [,1] [,2] [,3] [,4]
[1,]    1    2    3    4
[2,]    4    5    6    8
[3,]    7    8    9   11
[4,]   10   11   12   12

```

根據位置取得 matrix 上的資料

- 在 Console 上輸入下面敘述

`intRowMatrix[1]`

`intRowMatrix[2]`

`intRowMatrix[3,1]`

The screenshot shows the RStudio interface again. The terminal window displays the results of the three indexing operations:

```

> intRowMatrix[1]
[1] 1 2 3 4
> intRowMatrix[2]
[1] 4 5 6 8
> intRowMatrix[3,1]
[1] 7

```

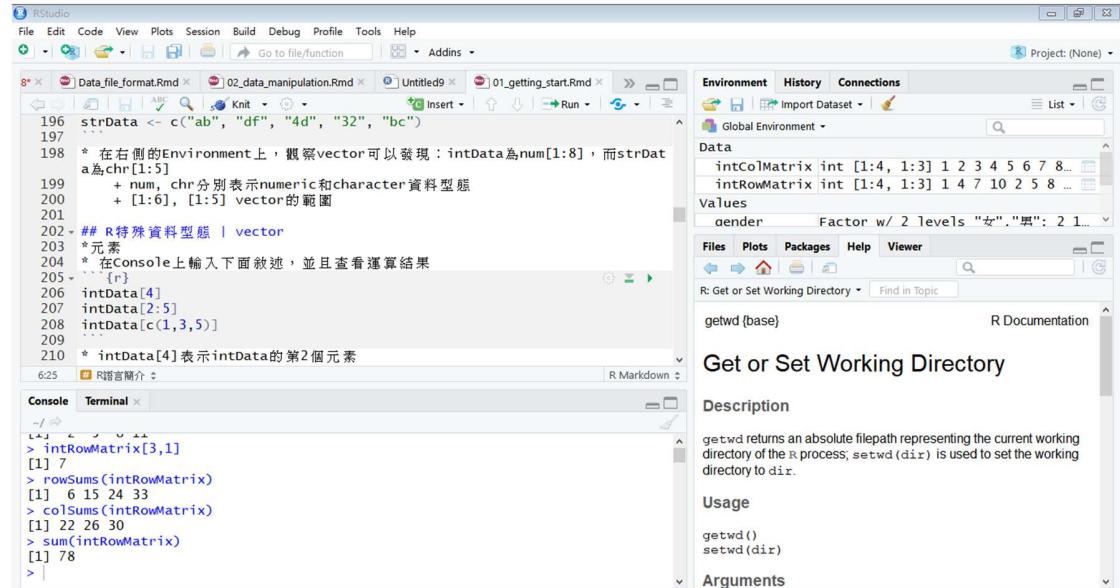
matrix 上的資料總和運算

- `rowSums(x)` : x 的每一橫列上的元素總和
- `colSums(x)` : x 的每一直行上的元素總和
- `sum(x)` : x 上的元素總和
- 在 Console 上輸入下面敘述

```
rowSums(intRowMatrix)
```

```
colSums(intRowMatrix)
```

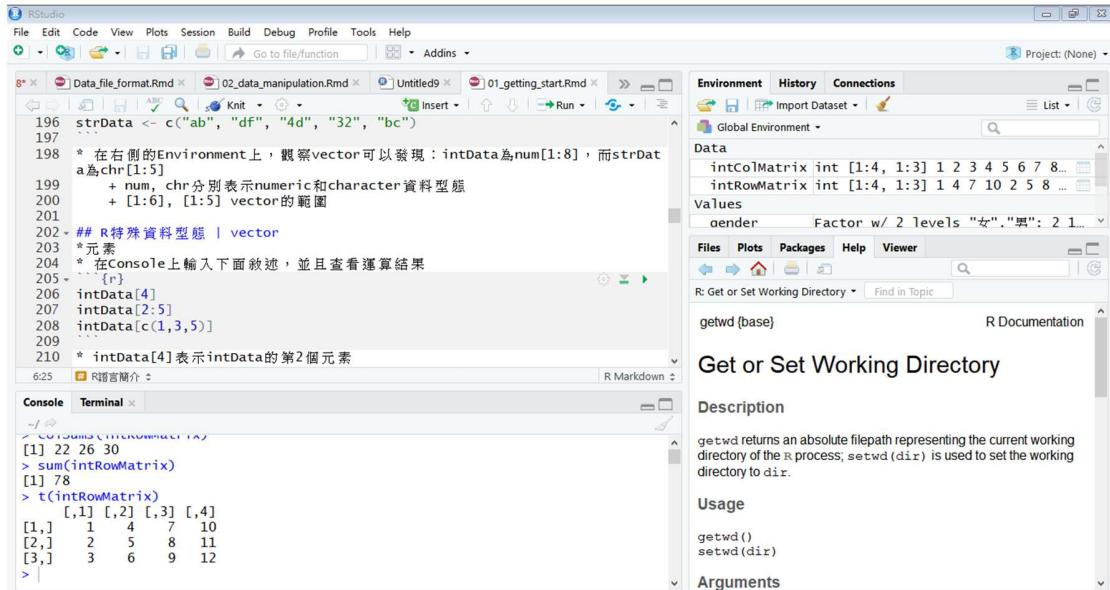
```
sum(intRowMatrix)
```



matrix 的轉置(transpose) 運算

- $t(x)$ ：轉置 x ，原先的行做為列，列做為行
- 在 Console 上輸入下面敘述

```
t(intRowMatrix)
```



matrix 的對角線(diagonal)運算

- `diag(x)` : x 的對角線
- 在 Console 上輸入下面敘述

```
diag(intRowMatrix)
```

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help, and Addins. The top right corner shows 'Project: (None)'. The left pane contains several Rmd files: Data_file_format.Rmd, 02_data_manipulation.Rmd, Untitled9, 01_getting_start.Rmd, and 01_getting_start.Rmd. The right pane has tabs for Environment, History, and Connections, with 'Environment' selected. It displays the Global Environment, Data, and Values sections. The Data section shows intColMatrix and intRowMatrix. The Values section shows aender. Below these are tabs for Files, Plots, Packages, Help, and Viewer. The bottom left is the R Markdown pane, and the bottom right is the R Documentation pane for the getwd function.

```
strData <- c("ab", "df", "4d", "32", "bc")
# 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData
# a為chr[1:5]
# + num, chr分別表示numeric和character資料型態
# + [1:6], [1:5] vector的範圍
## R 特殊資料型態 | vector
# 元素
# 在Console上輸入下面敘述，並且查看運算結果
intData[4]
intData[2:5]
intData[c(1,3,5)]
# intData[4] 表示intData的第2個元素
getwd(base)
R語言簡介
```

```
~/Documents/RMD
[1] 78
> t(intRowMatrix)
     [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
[3,]    3    6    9   12
> diag(intRowMatrix)
[1] 1 5 9
>
```

Description
getwd returns an absolute filepath representing the current working directory of the R process; setwd (dir) is used to set the working directory to dir.

Usage
getwd()
setwd(dir)

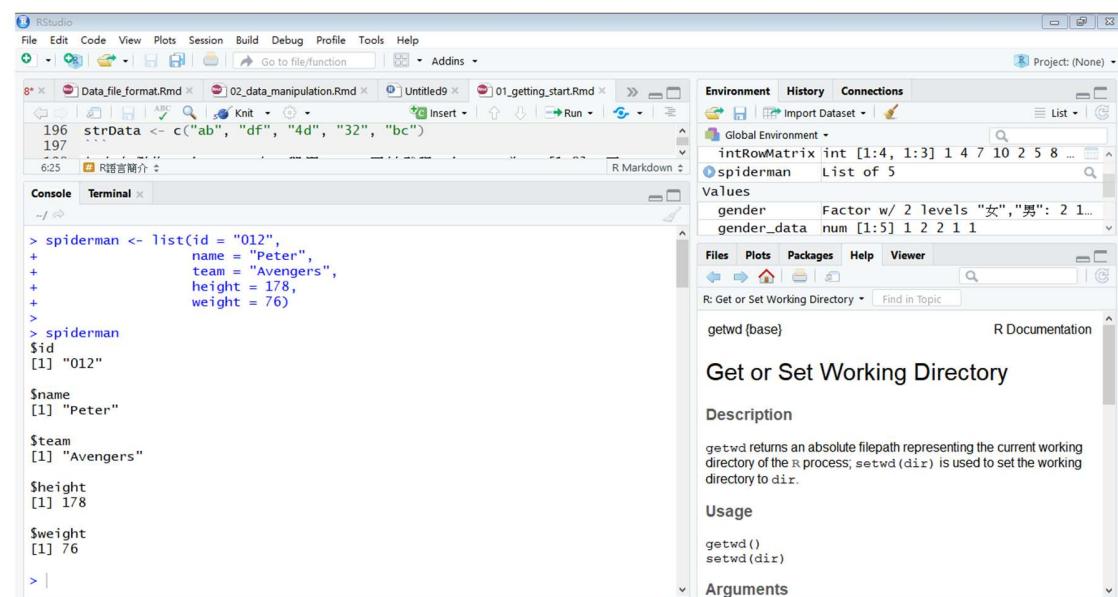
Arguments

list

- 可以包含不同資料型態的一序列資料
- 在 Console 上輸入下面敘述

```
spiderman <- list(id = "012",
                     name = "Peter",
                     team = "Avengers",
                     height = 178,
                     weight = 76)
```

spiderman



根據位置取得 list 上的資料

- 在 Console 上輸入下面敘述

```
spiderman[[3]]
```

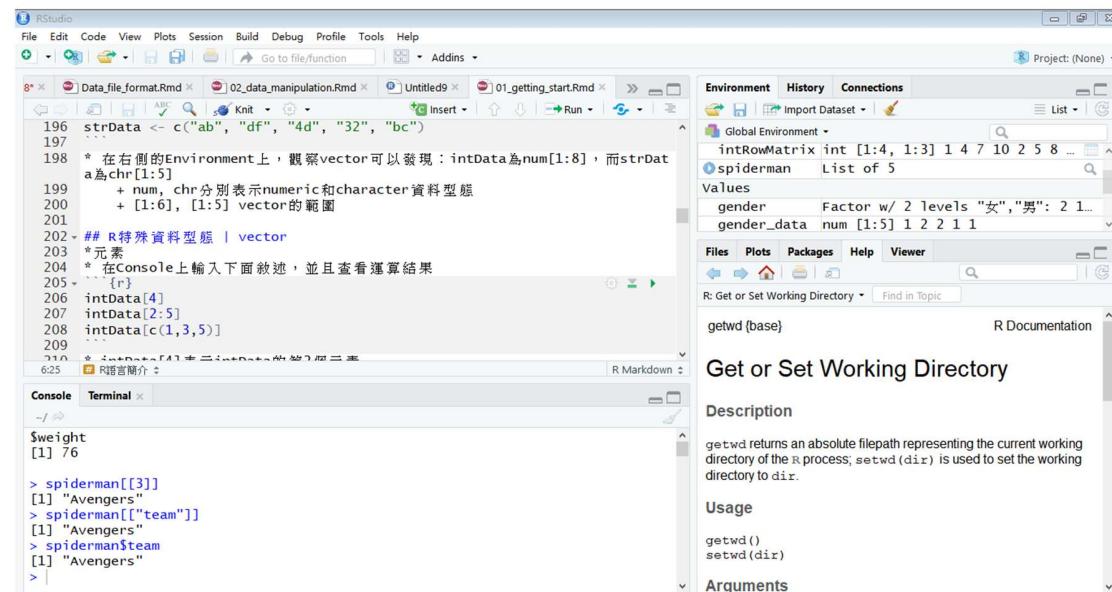
- 注意：在 vector 與 matrix 上，根據位置取得資料時只需一個中括號，但 list 需要使用兩個中括號

根據名稱取得 list 上的資料

- 在 Console 上輸入下面敘述

```
spiderman[["team"]]
```

```
spiderman$team
```

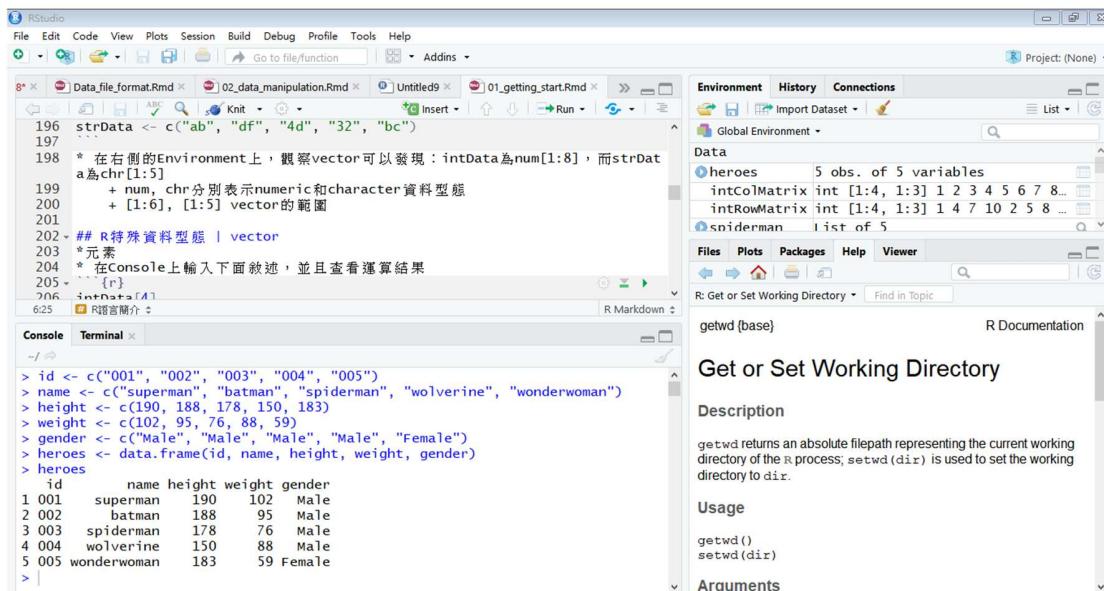


data frame

- 可以包含不同資料型態的資料集合
- 在 Console 上輸入下面敘述

```
id <- c("001", "002", "003", "004", "005")  
  
name <- c("superman", "batman", "spiderman", "wolverine", "wonderwoman")  
  
height <- c(190, 188, 178, 150, 183)  
  
weight <- c(102, 95, 76, 88, 59)  
  
gender <- c("Male", "Male", "Male", "Male", "Female")  
  
heroes <- data.frame(id, name, height, weight, gender)  
  
heroes
```

- data.frame 為 2 個維度的資料型態
- 橫列為 row，可以視為紀錄
- 直行為 column，可以視為資料欄位



The screenshot shows the RStudio interface with the following details:

- Console Tab:** Displays the R code and its output. The code defines variables for id, name, height, weight, and gender, and then creates a data frame named heroes. The output shows the contents of the heroes data frame.
- Environment Tab:** Shows the global environment with the heroes data frame and other objects like intData, num, chr, and spiderman.
- Help/Documentation:** A tooltip for the `getwd` function is displayed, providing information about getting or setting the working directory.

```
196 strData <- c("ab", "df", "4d", "32", "bc")  
197  
198 * 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData  
a為chr[1:5]  
199 + num, chr分別表示numeric和character資料型態  
200 + [1:6], [1:5] vector的範圍  
201  
202 ## R特殊資料型態 | vector  
203 * 元素  
204 * 在Console上輸入下面敘述，並且查看運算結果  
205 {  
206 intData[1]  
625 R語言簡介
```

	id	name	height	weight	gender
1	001	superman	190	102	Male
2	002	batman	188	95	Male
3	003	spiderman	178	76	Male
4	004	wolverine	150	88	Male
5	005	wonderwoman	183	59	Female

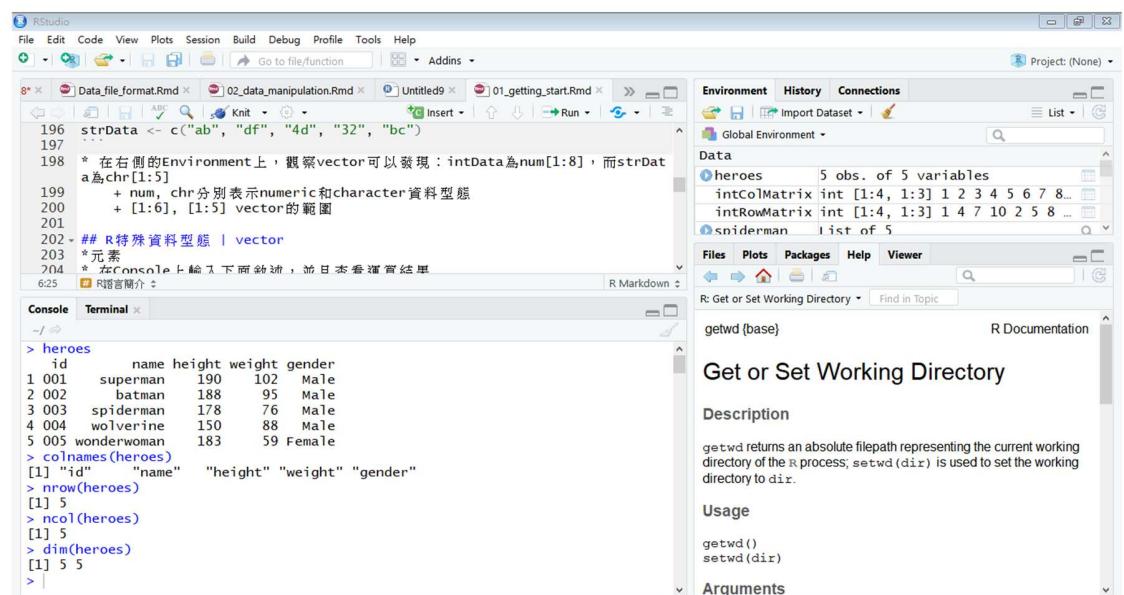
data frame 的資料數量

- 在 Console 上輸入下面敘述

```
nrow(heroes)
```

```
ncol(heroes)
```

```
dim(heroes)
```



根據位置取得 data frame 上的資料

- 在 Console 上輸入下面敘述

```
heroes[2, ]
```

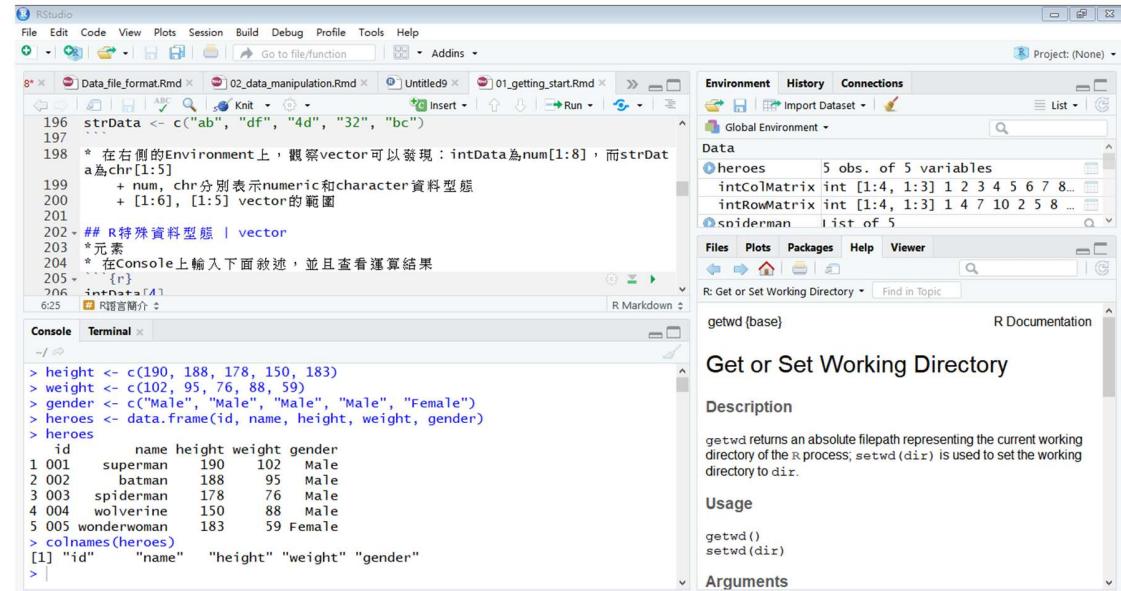
```
heroes[, 2]
```

```
heroes[2, 2]
```

data frame 上的資料欄位名稱

- 在 Console 上輸入下面敘述

```
colnames(heroes)
```



根據名稱取得 data frame 上的欄位資料

- 在 Console 上輸入下面敘述

```
heroes[["name"]]
```

```
heroes$name
```

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** Displays R code from multiple files. The visible code includes:


```

196 strData <- c("ab", "df", "4d", "32", "bc")
197
198 * 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData
  a為chr[1:5]
199   + num, chr分別表示numeric和character資料型態
200   + [1:6], [1:5] vector的範圍
201
202 ## R特殊資料型態 | vector
203 * 元素
204 * 在Console上輸入下面敘述，並且查看運算結果
205 {r}
206 intData[4]
207 intData[2:5]
208 intData[1:3, 5:7]
      
```
- Environment View:** Shows the global environment with objects like `heroes`, `intColMatrix`, `intRowMatrix`, and `spiderman`.
- Console View:** Shows the R terminal session output. The user has run several commands to inspect the `heroes` dataset, including printing rows 2 and 2:2, extracting column names, and printing the first few rows.
- Help View:** A tooltip for the `getwd` function is displayed, providing its usage and arguments.

Get or Set Working Directory

Description

`getwd` returns an absolute filepath representing the current working directory of the R process; `setwd(dir)` is used to set the working directory to `dir`.

Usage

```
getwd()
setwd(dir)
```

Arguments

根據索引取得 data frame 上的紀錄資料

- 索引 | : data frame 上符合條件的紀錄
- 在 Console 上輸入下面敘述

```
heroes[heroes$name=="spiderman",]
```

- 在 Console 上輸入下面敘述，查看索引

```
heroes$name=="spiderman"
```

The screenshot shows the RStudio interface. On the left, there are several Rmd files open in tabs. In the center, the Environment pane displays the 'heroes' data frame and a 'spiderman' list object. On the right, the Help pane is open for the 'getwd' function, showing its description, usage, and arguments.

Code in the Rmd files:

```
196 strData <- c("ab", "df", "4d", "32", "bc")
197
198 * 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData
  a為chr[1:5]
199   + num, chr分別表示numeric和character資料型態
200   + [1:6], [1:5] vector的範圍
201
202 ## R特殊資料型態 | vector
203 * 元素
204 * 在Console上輸入下面敘述，並且查看運算結果
205 {r}
206 intData[4]
207 intData[2:5]
208 intData[1:3, 5:6]
209
210 #> 請看簡介
```

Console output:

```
> heroes[2, 2]
[1] "batman"
> heroes[["name"]]
[1] "superman"    "batman"      "spiderman"   "wolverine"   "wonderwoman"
> heroes$name
[1] "superman"    "batman"      "spiderman"   "wolverine"   "wonderwoman"
> heroes[heroes$name=="spiderman",]
#>   id   name height weight gender
#> 3 003 spiderman 178    76 Male
> heroes$name=="spiderman"
[1] FALSE FALSE  TRUE FALSE FALSE
>
```

練習

- 取得性別為女性的超級英雄
- 取得身高高於 180cm 的超級英雄

根據索引取得 data frame 上的資料

- 在 Console 上輸入下面敘述

```
heroes[which.max(heroes$height),]
```

- 在 Console 上輸入下面敘述，查看索引

```
which.max(heroes$height)
```

- 注意：which.max(x)：傳回 x 中最大值的位置

The screenshot shows the RStudio interface. In the top-left pane, there are several R script files listed. In the bottom-left pane, the 'Console' tab is active, showing R code and its output. The output includes:

```
strData <- c("ab", "df", "4d", "32", "bc")
#> [1] "ab" "df" "4d" "32" "bc"
#> 
#> * 在右側的Environment上，觀察vector可以發現：intData為num[1:8]，而strData
#> a為chr[1:5]
#>     + num, chr分別表示numeric和character資料型態
#>     + [1:6], [1:5] vector的範圍
#> 
#> ## R特殊資料型態 | vector
#> * 元素
#> * 在Console上輸入下面敘述，並且查看運算結果
#> 
#> intData[4]
#> intData[2:5]
#> intData[1:3, 5]
#> 
#> #> 請看简介
```

In the bottom-right pane, the 'Help' tab is active, displaying the documentation for the 'getwd' function.

Get or Set Working Directory

Description

`getwd` returns an absolute filepath representing the current working directory of the R process; `setwd(dir)` is used to set the working directory to `dir`.

Usage

```
getwd()
setwd(dir)
```

Arguments

練習

- 取得體重最重的超級英雄
- 取得體重最輕的超級英雄
- 取得名字最長的超級英雄

本次課程小結

小結

- R 語言中很重要的是使用函數來幫忙完成工作
- 在本次課程中，列舉出許多常用的函數
- 但還有更多的函數
- 面對這些函數，死背下來不是最好的方法
- 需要學習如何查找到適合的函數是學習 R 語言的訣竅

小結

- 在 R 語言的程式設計當中，vector 與 data frame 是相當重要的資料型態
- 許多分析與計算都在 vector 或 data frame 上完成
- 下一節將主要著重在 data frame 為主的資料讀取與儲存

延伸思考

1. 為什麼要學程式設計？為什麼要學 R 語言的程式設計？
2. 有沒有學程式設計的訣竅？