# 第 05 天

## 把變數集結起來

在第五天要開始討論資料結構，為了不要讓讀者被標題嚇跑，我們刻意沒有直接把**資料結構**當作章節的名稱，實際上我們要探討的內容跟您印象中很艱澀的資料結構是沒有什麼相關的，我們只是很單純地想讓您知道 R 語言可以用哪些方法將變數收藏起來。比如說我們想要在 R 語言中輸入四個季節的名稱：

season\_1 <- "spring"  
season\_2 <- "summer"  
season\_3 <- "autumn"  
season\_4 <- "winter"

這是到目前為止我們儲存變數的方式，將一個文字對應儲存給一個物件之中。我們打算用三天的篇幅介紹 R 語言三大類型的資料結構，今天就從一維這個大類型開始說起：

* **一維**
  + **向量（vector）**
  + **因素向量（factor）**
* 二維
  + 矩陣（matrix）
  + 資料框（data.frame）
* 多維
  + 陣列（array）
  + 清單（list）

### 向量

我們也可以將四季的名稱放置在一個好像有四個格子的抽屜中：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> four\_seasons  
[1] "spring" "summer" "autumn" "winter"

這就是一個由文字組成的**向量（Vector）**，這是一個我們會很常使用的資料結構，使用 c() 函數將我們想要放入的任意變數集結在一個向量之中：

> weekends <- c("Saturday", "Sunday")  
> my\_lucky\_numbers <- c(7L, 24L)  
>   
> weekends  
[1] "Saturday" "Sunday"   
> my\_lucky\_numbers  
[1] 7 24

我們提過向量就像是有格子的抽屜，每個格子上面都有一個索引值，方便 R 語言搜尋放在裡頭的變數，舉例說假如我們最喜歡的季節是秋天，可以使用 [3] 這個索引值選出秋天：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> favorite\_season <- four\_seasons[3]  
> favorite\_season  
[1] "autumn"

或者我們認為夏天太熱、冬天太冷，不要放在喜歡的季節中，可以使用 [c(-2, -4)] 刪去：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> favorite\_seasons <- four\_seasons[c(-2, -4)]  
> favorite\_seasons  
[1] "spring" "autumn"

向量有一個很特別的屬性，那就是只能包含**一種**變數類型，舉例來說，如果我們將整數跟數值放入同一個向量之中，那麼向量會將整數轉換成數值：

> lucky\_numbers <- c(7L, 24)  
> class(lucky\_numbers[1])  
[1] "numeric"

如果我們將邏輯值跟整數放入同一個向量之中，那麼向量會將邏輯值轉換成整數：

> lucky\_numbers <- c(7L, FALSE)  
> lucky\_numbers  
[1] 7 0  
> class(lucky\_numbers[2])  
[1] "integer"

假如是將邏輯值、整數、數值還有文字放入同一個向量之中，那麼向量會全部轉換成文字：

> mixed\_vars <- c(TRUE, 7L, 24, "spring")  
> mixed\_vars  
[1] "TRUE" "7" "24" "spring"  
> class(mixed\_vars[1])  
[1] "character"  
> class(mixed\_vars[2])  
[1] "character"  
> class(mixed\_vars[3])  
[1] "character"

除了直接使用索引值從向量選出變數，還能夠利用我們在[第 02 天：暸解不同的變數類型](chapter2.md)提到的**判斷運算子**來對向量進行篩選：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> my\_favorite\_season <- four\_seasons == "autumn"  
> four\_seasons[my\_favorite\_season]  
[1] "autumn"

如果不只一個條件，我們可以使用 **&（AND）** 以及 **|（OR）** 這兩個符號連結判斷條件：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> my\_favorite\_seasons <- four\_seasons == "spring" | four\_seasons == "autumn" # 我喜歡春天或秋天  
> four\_seasons[my\_favorite\_seasons]  
[1] "spring" "autumn"

在 R 語言中產生向量的方式除了使用 c() 函數手動建立以外，還可以透過 rep() 函數、seq() 函數與 : 快速產生。

rep() 函數可以生成重複變數的向量，times 參數可以指定要生成幾個：

> rep(7L, times = 8)  
[1] 7 7 7 7 7 7 7 7  
> rep("R", times = 10)  
 [1] "R" "R" "R" "R" "R" "R" "R" "R" "R" "R"

seq() 函數可以生成等差數列，from 參數指定數列的起始值，to 參數指定數列的終止值，by 參數指定數值的間距：

> seq(from = 7, to = 77, by = 7)  
 [1] 7 14 21 28 35 42 49 56 63 70 77  
> seq(from = 1, to = 10, by = 1)  
 [1] 1 2 3 4 5 6 7 8 9 10

如果單純是要生成數值間距為 1 的數列，用 : 更快捷：

> 1:10  
 [1] 1 2 3 4 5 6 7 8 9 10  
> 11:20  
 [1] 11 12 13 14 15 16 17 18 19 20

### 因素向量

R 語言針對儲存文字的向量有一個特殊的資料結構稱為因素向量（factor），它是一個帶有層級（Levels）資訊的向量，我們使用 factor() 函數可以將向量轉換成因素向量：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> four\_seasons  
[1] "spring" "summer" "autumn" "winter"  
> four\_seasons\_factor <- factor(four\_seasons)  
> four\_seasons\_factor  
[1] spring summer autumn winter  
Levels: autumn spring summer winter

眼尖的您有發現輸出因素向量時會將層級的資訊輸出嗎？接下來我們在轉換為因素向量時利用 ordered = TRUE 與 levels = 的參數來排列一下對於四個季節的喜好：

> four\_seasons <- c("spring", "summer", "autumn", "winter")  
> four\_seasons\_factor <- factor(four\_seasons, ordered = TRUE, levels = c("summer", "winter", "spring", "autumn"))  
> four\_seasons\_factor  
[1] spring summer autumn winter  
Levels: summer < winter < spring < autumn

您有發現這時輸出的因素向量除了層級，還有大小的關係資訊嗎？通常有排序的因素向量適合用在有順序的文字，像是形容溫度的文字：

> temperatures <- c("warm", "hot", "cold")  
> temp\_factors <- factor(temperatures, ordered = TRUE, levels = c("cold", "warm", "hot"))  
> temp\_factors  
[1] warm hot cold  
Levels: cold < warm < hot

假如我們只有指定 ordered = TRUE 但沒有指定 levels =，R 語言會預設使用字母順序排序，可能就會產生不符合直覺的排序，像是：

> temperatures <- c("warm", "hot", "cold")  
> temp\_factors <- factor(temperatures, ordered = TRUE)  
> temp\_factors  
[1] warm hot cold  
Levels: cold < hot < warm

### 小結

好啦！第五天的內容就是這麼多，我們認識了 R 語言的兩種資料結構：向量與因素向量。

### 練習題

###### 1. 我們有一個文字向量 weekdays 是一週的星期一到星期五，請您將最喜歡的週五（Happy Friday）從這個向量中用索引值選出來並且指派給 favorite\_day

weekdays <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")  
favorite\_day <- weekdays[\_\_\_]  
favorite\_day

###### 2. 同樣的文字向量，請您利用**判斷運算子**將最藍的週一（Blue Monday）從這個向量中剔除後將剩餘的日子指派給 without\_monday

weekdays <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")  
not\_blue\_monday <- weekdays != "Monday"  
without\_monday <- weekdays[\_\_\_]  
without\_monday

###### 3. 我們有一個文字向量 speed\_char 描述速度的快慢，請您使用 factor() 函數轉換成因素向量 speed\_factor，並且讓快慢有排序（慢 < 快）

speed\_char <- c("slow", "fast")  
speed\_factor <- \_\_\_(speed\_char, ordered = \_\_\_, levels = c("\_\_\_", "\_\_\_"))  
speed\_factor

### 延伸資訊

* [R 語言導論](https://www.datacamp.com/community/open-courses/r-%E8%AA%9E%E8%A8%80%E5%B0%8E%E8%AB%96#gs.29lL1x4)
* [Introduction to R](https://www.datacamp.com/courses/free-introduction-to-r)