# 第 03 天

## 暸解不同的變數類型（2）

第三天會討論剩餘的三種變數類型：文字、日期與時間，我們再複習一下第二天一開始出現的表格，今天會說明的我們以**粗體**表示：

|  |  |  |
| --- | --- | --- |
| 變數類型 | 變數類型英文 | 範例 |
| 數值 | numeric | 2 |
| 整數 | integer | 2L |
| 邏輯值 | logical | TRUE |
| **文字** | character | “Learning R the easy way” |
| **日期** | Date | Sys.Date() |
| **時間** | POSIXct POSIXt | Sys.time() |

### 文字

在 R 語言中我們可以使用單引號（’）或雙引號（“）來建立文字（character），我的習慣是使用雙引號（”），所以在本書中的範例都會使用 " 來建立文字。

> first\_name <- 'Tony'  
> first\_name  
[1] "Tony"  
> class(first\_name)  
[1] "character"

> first\_name <- "Tony"  
> first\_name  
[1] "Tony"  
> class(first\_name)  
[1] "character"

### 日期

在 R 語言中被定義為日期（Date）的變數外觀看起來跟文字沒有什麼差別，但是我們一但將它們放入 class() 函數中檢驗，就會發現它並不是文字，我們接下來要使用的 Sys.Date() 是一個不需要任何輸入就會輸出電腦系統日期的函數。

> sys\_date <- Sys.Date() # 系統日期  
> sys\_date # 看起來跟文字相同  
[1] "2017-02-24"  
> class(sys\_date)  
[1] "Date"

而這兩個變數類型最大的分野，就在於日期是可以被轉換為整數，而文字是不行的：

> sys\_date <- Sys.Date()  
> sys\_date\_char <- as.character(sys\_date) # 創造一個文字類型  
>   
> as.integer(sys\_date)  
[1] 17221  
> as.integer(sys\_date\_char)  
[1] NA  
Warning message:  
NAs introduced by coercion

這裡我們使用的 as.character() 與 as.integer() 函數是 R 語言中用來進行變數類型轉換的函數，我們可以清楚文字類型的系統日期在轉換整數時失敗，以致於產生一個遺失值（Not Available，NA）。

那麼日期類型的系統日期轉換的整數是有什麼根據嗎？答案是有的，R 語言預設以西元 1970 年 1 月 1 日作為 0，在這一天以後的每天都 +1 來記錄，而這一天以前的每天都 -1 來記錄。

> date\_of\_origin <- as.Date("1970-01-01")  
>   
> as.integer(date\_of\_origin)  
[1] 0  
> as.integer(date\_of\_origin + 1)  
[1] 1  
> as.integer(date\_of\_origin - 1)  
[1] -1  
> date\_of\_origin  
[1] "1970-01-01"  
> date\_of\_origin + 1  
[1] "1970-01-02"  
> date\_of\_origin - 1  
[1] "1969-12-31"

也因為這樣的特性，日期類型可以進行四則運算，而文字類型不行：

> sys\_date <- Sys.Date()  
> sys\_date\_char <- as.character(sys\_date) # 創造一個文字類型  
>   
> sys\_date - 1 # 昨天的日期  
[1] "2017-02-23"  
> sys\_date\_char - 1  
Error in sys\_date\_char - 1 : non-numeric argument to binary operator

### 時間

在 R 語言中被定義為時間（POSIXct POSIXt）的變數外觀看起來跟文字同樣也沒有什麼差別，但是我們一但將它們放入 class() 函數中檢驗，就會發現它並不是文字，我們接下來要使用的 Sys.time() 是一個不需要任何輸入就會輸出電腦系統時間的函數。

> sys\_time <- Sys.time() # 系統時間  
> sys\_time # 看起來跟文字相同  
[1] "2017-02-24 16:06:50 CST"  
> class(sys\_time)  
[1] "POSIXct" "POSIXt"

與日期相似，時間亦是可以被轉換為整數的：

> sys\_time <- Sys.time()  
> as.integer(sys\_time)  
[1] 1487923726

與日期相似，R 語言預設以西元 1970 年 1 月 1 日格林威治標準時間（Greenwich Mean Time，GMT）00 時 00 分 00 秒作為 0，在這個時間點以後的每秒都 +1 來記錄，這個時間點以前的每秒都 -1 來記錄。

> time\_of\_origin <- as.POSIXct("1970-01-01 00:00:00", tz = "GMT")  
> as.integer(time\_of\_origin)  
[1] 0  
> as.integer(time\_of\_origin + 1)  
[1] 1  
> as.integer(time\_of\_origin - 1)  
[1] -1  
> time\_of\_origin  
[1] "1970-01-01 GMT"  
> time\_of\_origin + 1  
[1] "1970-01-01 00:00:01 GMT"  
> time\_of\_origin - 1  
[1] "1969-12-31 23:59:59 GMT"

我們這裡所使用的參數 tz = "GMT" 是指定時區，假如您的電腦和我一樣時區是設在中原標準時間（Chungyuan Standard Time，CST），晚格林威治標準時間八個小時（GMT + 8），則基準時間會是西元 1970 年 1 月 1 日 08 時 00 分 00 秒。

> time\_of\_origin\_cst <- as.POSIXct("1970-01-01 08:00:00")  
> as.integer(time\_of\_origin\_cst)  
[1] 0

### 小結

第三天我們再學會了三種 R 語言的基本變數類別：文字（character）、日期（Date）與時間（POSIXct POSIXt）。

### 練習題

###### 1. 將您的姓名指派給 my\_name 並且輸出在命令列

my\_name <- "\_\_\_"  
my\_name

###### 2. 香港搖滾樂隊 Beyond 於 1983 年成立，我們假設成立日期是 1983-12-31，請將成立日期指派給 beyond\_start 並轉換成整數輸出在命令列

beyond\_start <- as.Date("\_\_\_")  
as.integer(\_\_\_)

###### 3. 請以系統日期計算今年是 Beyond 成立幾週年紀念？

beyond\_start <- as.Date("\_\_\_")  
days\_diff <- \_\_\_ - \_\_\_ # 計算天數差距  
years\_diff <- \_\_\_ / 365 # 除以 365 換算成年

### 延伸資訊

* [An Introduction to R](https://cran.r-project.org/doc/manuals/r-release/R-intro.html)