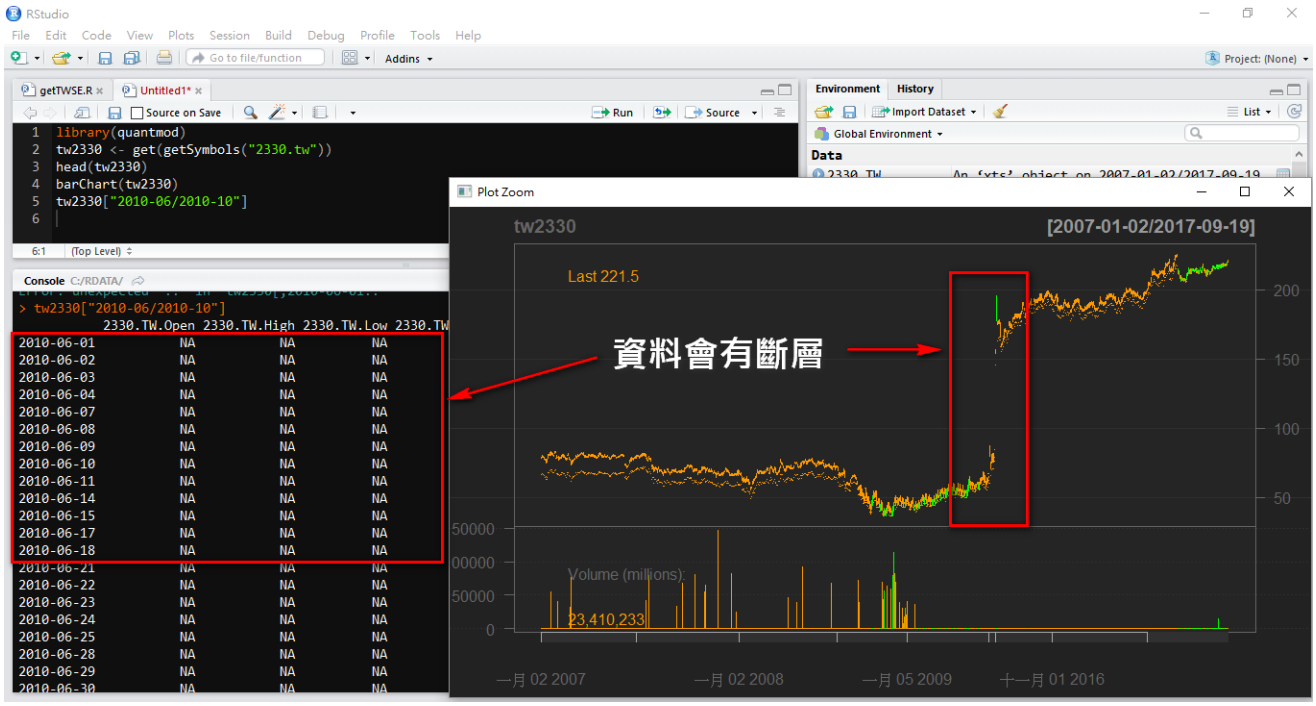
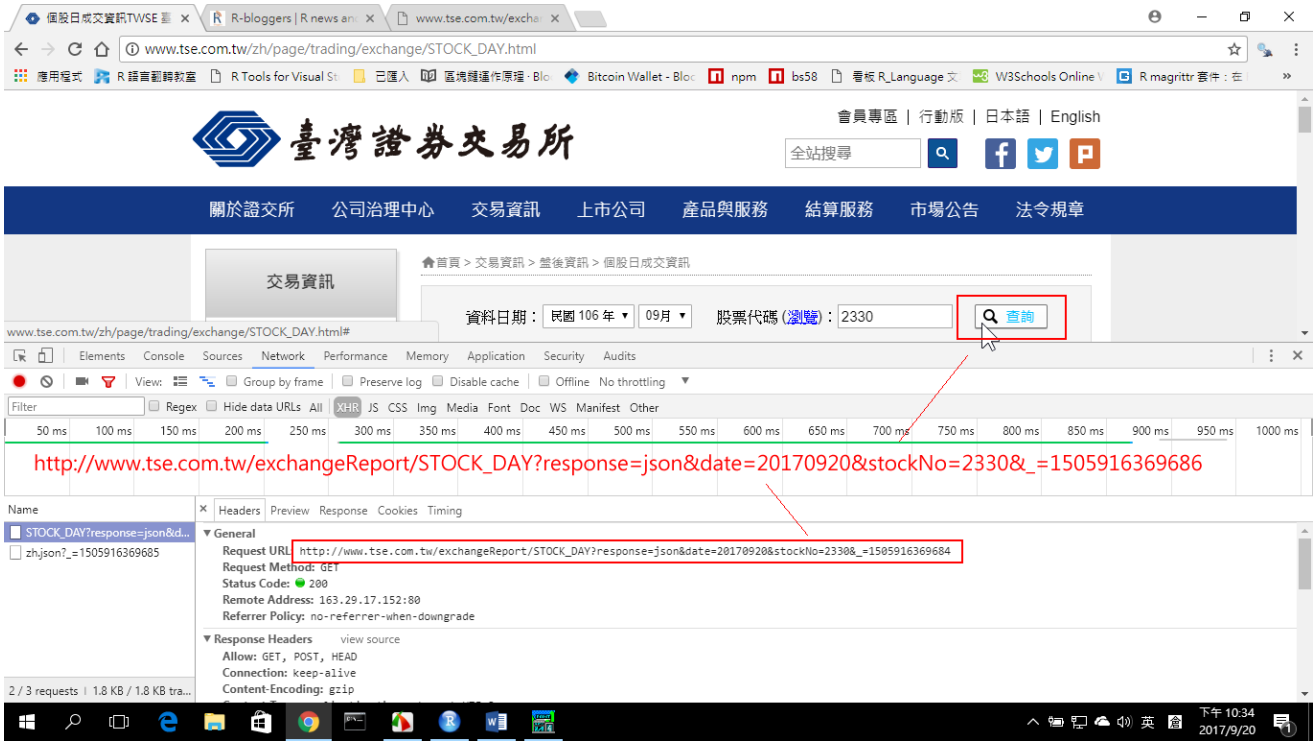
自己寫個 Function 抓證交所的股票資訊

金融上的套件: quantmod，可在 Yahoo finance、Googlefinance 等網站，下載公開數據進行分析，但對於台股而言，雖然可在 Yahoo finance 抓取，但有時資料並不完整。
以台積電(2330)為例，透過 getSymbols("2330.tw") 抓取回來的資料，發現有缺漏。



因此改抓台灣的證券交易所，就是我們的目標。
網路上有許多類似的方法，可抓取證交所的股票資料，在這裏我也嚐試自己寫一個 Function
首先是利用 Google Chrome 開啟證交所網頁，點[交易資訊] -> [盤後資訊] -> [個股日成交資訊]

按下 F12，可進入開發人員工具模式，有利我們觀查網頁的運作，以下圖為例
當按下[查詢]後，可發現網頁以 GET 方式，送出一串 URL，仔細觀查可發現 URL 中帶有個參數，其中 respons=json，代表回傳格式為 JSON，date=20170920，代表查詢日期，實測結果發現，日期的最後 2 碼並不影響查詢結果，即 date=20170901 與 date=20170920 查回來的結果是相同的，稍後的自訂函數中，天數統一用 01，stockNo=2330 為股票代號，最後一個參數 _=1505916369686，為時間戳記，隨時間變動。



回傳 JSON


```
{"stat":"OK","date":"20170920","title":"106年09月 2330 台積電　　各日成交資訊","fields":["日期","成交股數","成交金額","開盤價","最高價","最低價","收盤價","漲跌價差","成交筆數"],"data":[["106/09/01","17,805,405","3,851,834,126","215.00","217.50","215.00","216.50"," 0.00","6,956"],
["106/09/04","15,053,185","3,273,164,145","216.50","218.00","215.50","217.50","+1.00","8,635"],
["106/09/05","34,514,275","7,503,488,603","218.00","218.00","216.50","218.00","+0.50","7,065"],
["106/09/06","28,148,528","6,117,444,609","216.50","218.00","216.00","217.00","-1.00","6,999"],
["106/09/07","19,198,540","4,170,321,680","218.00","218.50","216.00","217.00"," 0.00","6,586"],
["106/09/08","25,512,045","5,549,520,265","217.00","218.00","216.50","218.00","+1.00","7,969"],
["106/09/11","19,387,886","4,215,723,762","219.00","219.50","216.50","217.00","-1.00","5,305"],
["106/09/12","24,862,321","5,437,521,978","219.00","219.50","217.50","219.00","+2.00","5,992"],
["106/09/13","18,336,541","3,998,984,438","217.50","219.00","217.50","218.00","-1.00","4,598"],
["106/09/14","21,508,567","4,701,852,606","218.50","219.00","217.50","218.50","+0.50","6,314"],
["106/09/15","48,901,103","10,668,230,939","218.50","219.00","217.00","218.50"," 0.00","5,253"],
["106/09/18","17,812,705","3,905,801,892","219.00","220.00","218.00","219.50","+1.00","6,714"],
["106/09/19","26,255,575","5,742,129,525","221.50","221.50","220.00","221.50","+2.00","11,072"],
["106/09/20","23,434,237","5,179,958,449","221.00","222.00","220.00","221.50"," 0.00","9,347"]],"notes":["符號說明:+/-/X表示漲/跌/不比價","當日統計資訊含一般、零股、盤後定價、鉅額交易，不含拍賣、標購。","ETF證券代號第六碼為K、M、S、C者，表示該ETF以外幣交易。"]}
```


有了上述的基本觀念，將有利於 Function 的寫作。

函數：getTWSE(stockNo, from = c(year, month), to = c(year, month))，預設有 3 個參數，皆帶有預設值；當呼叫函數而未帶參數時，系統會用預設值當查詢條件。

stockNo = “2330”，預設為台積電(2330)

```
24 # ============================================================================================================
25 getTWSE <- function(stockNo = "2330",
26                     from = c(as.integer(format(Sys.Date(),"%Y")), as.integer(format(Sys.Date(),"%m"))),
27                     to = c(as.integer(format(Sys.Date(),"%Y")), as.integer(format(Sys.Date(),"%m")))) {
```

From、to，若未輸入，預設為查詢當下年月，如下圖

```
Console C:/RDATA/
> from = c(as.integer(format(Sys.Date(),"%Y")), as.integer(format(Sys.Date(),"%m")))
> to = c(as.integer(format(Sys.Date(),"%Y")), as.integer(format(Sys.Date(),"%m")))
> from
[1] 2017    9          <- from
> to
[1] 2017    9          <- to
>
```

行 38~45，主要是檢查 RStudio 是否已載入 jsonlite 套件，未載入則印出訊息並中斷 Function，也可以改寫成未載入，則自動安裝載入套件

```
38     # ============================================================================================================
39     # Using this function, you must install.packages("jsonlite") and library(jsonlite)
40     # ============================================================================================================
41     packages <- gsub("package:", replacement = "", search())
42     if (!("jsonlite" %in% packages)){
43         print("Error: The 'jsonlite' package has not been loaded.")
44         return(NULL)
45     }
```

改為偵測到未載入則進行安裝及載入動作

```
1  # ============================================================================================================
2  # Using this function, you must install.packages("jsonlite") and library(jsonlite)
3  # ============================================================================================================
4  packages <- gsub("package:", replacement = "", search())
5  if (!("jsonlite" %in% packages)){
6    install.packages("jsonlite")
7    library(jsonlite)                <- 安裝並載入
8  }
```

行 61~64，判斷 from 及 to 的內容，若僅有年度，則補上月份

```
47     # ============================================================================================================
48     # parameter check & parse
49     # ============================================================================================================
50     if (from[1] > to[1]){
51         print("The starting year is greater than the deadline.")
52         return(NULL)
53     }
54     nowYear <- as.integer(format(Sys.Date(),"%Y"))
55     nowMonth <- as.integer(format(Sys.Date(),"%m"))
56     MM <- c("01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12")
57     queryDate <- NULL                  當from只有年, 補為c(年, 1)
58     historyStock <- NULL               當to只有年, 若為當年度, 則月份到當下月
59     startYM <- from                    若非當年, 則c(年, 12)
60     endYM <- to
61     if (length(startYM) == 1) startYM <- c(startYM, 1)
62     if (length(endYM) == 1){
63         ifelse (endYM[1] == nowYear, endYM <- c(endYM, nowMonth), endYM <- c(endYM, 12))
64     }
65     if (startYM[2] < 1 | startYM[2] > 12 | endYM[2] < 1 | endYM[2] > 12){
66         print("Month must be between 1 ~ 12.")
67         return(NULL)
68     }
```

行 73~85，計算出查詢區間的明細，待會組 URL 時，參數 date 的內容

```
72      # ============================================================================
73      if (startYM[1] == endYM[1]) {
74          queryDate <- paste0(startYM[1], MM[startYM[2]:endYM[2]], "01")
75      } else if ((endYM[1] - startYM[1]) == 1) {
76          queryDate <- c(paste0(startYM[1], MM[startYM[2]:12], "01"),
77                         paste0(endYM[1], MM[1:endYM[2]], "01"))
78      } else {
79          tmpY <- c((startYM[1]+1):(endYM[1]-1))
80          queryDate <- paste0(startYM[1], MM[startYM[2]:12], "01")
81          for (tY in tmpY){
82              queryDate <- c(QueryDate, paste0(tY, MM, "01"))
83          }
84          queryDate <- c(QueryDate, paste0(endYM[1], MM[1:endYM[2]], "01"))
85      }
```

行 91，組查詢網址，傳入 4 個參數；行 92，真正到證交所抓資料

```
86      # ============================================================================
87      # to TWSE get History Stock
88      # ============================================================================
89      for (qyDate in queryDate){
90          ttime <- as.character(as.integer(as.POSIXct(Sys.time()))*100)
91          twseUrl <- paste0(url, "response=", response, "&date=", qyDate, "&stockNo=", stockNo, "& =", ttime)
92          jsonData <- fromJSON(twseUrl, flatten = TRUE)
93          if (jsonData$stat == "OK"){
94              tmpStock <- data.frame(jsonData$data[, 1],
95                                     jsonData$data[, 4:7],
96                                     jsonData$data[, 2:3],
97                                     stringsAsFactors = FALSE)
98              historyStock <- rbind(historyStock, tmpStock)
99          }
100     }
```

組查詢網址

抓取JSON

行 109~117，資料格式轉換，行 111，利用自訂函數 CNV_DATE()，透過 sapply 函數將民國年轉為西元年，行 112~117，則是將 string 轉為 number，因為含千分號(,)，故先要濾掉

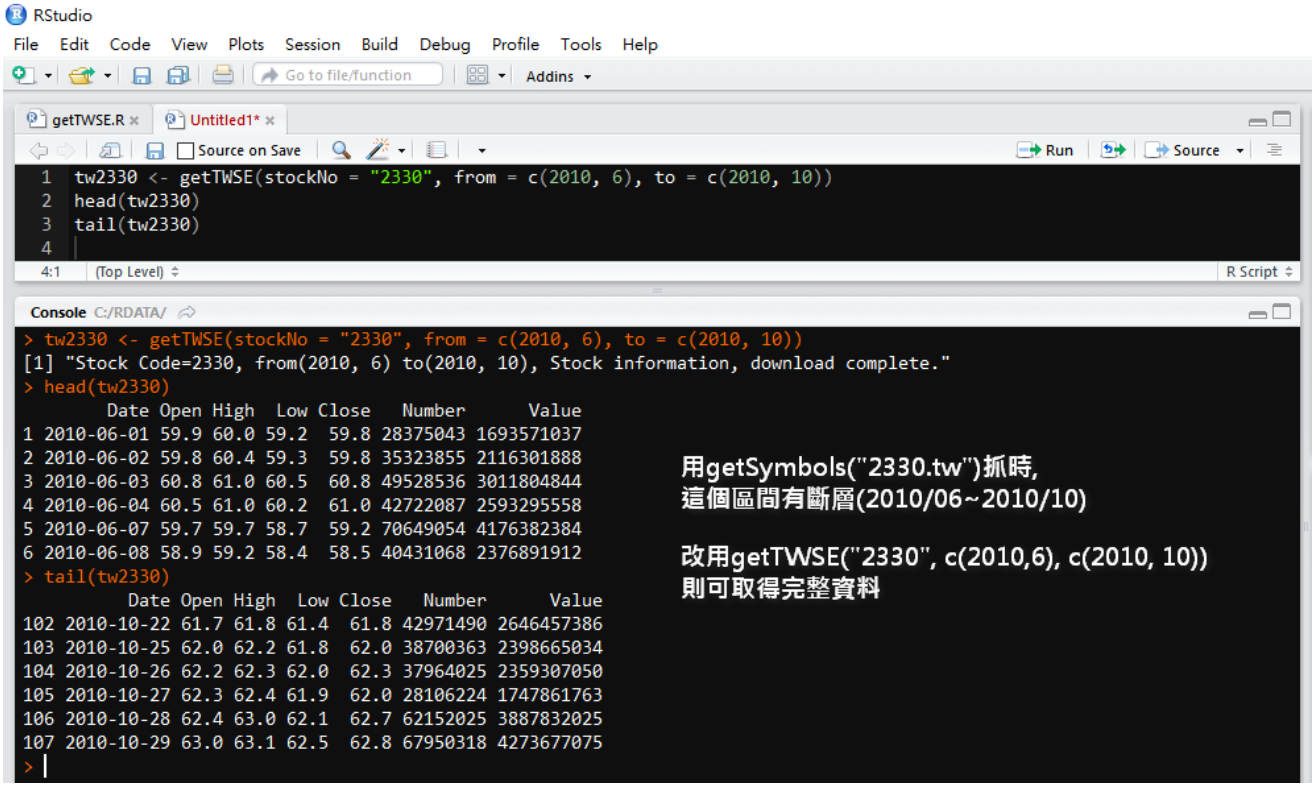
```
101     # ============================================================================
102     # Convert Data Format:
103     # If the string has a comma, it can not be converted to a value.
104     # gsub(',', replacement = '', x) --> remove comma
105     #
106     # Ex. as.numeric("196,857,432") --> Warning message: NAs introduced by coercion
107     #     as.numeric("196857432") --> 196857432
108     # ============================================================================
109     if (!is.null(historyStock)){
110         colnames(historyStock) <- c("Date", "Open", "High", "Low", "Close", "Number", "Value")
111         historyStock$Date <- sapply(historyStock$Date, CNV_DATE)
112         historyStock$Open <- as.numeric(gsub(',', replacement = '', historyStock$Open))
113         historyStock$High <- as.numeric(gsub(',', replacement = '', historyStock$High))
114         historyStock$Low <- as.numeric(gsub(',', replacement = '', historyStock$Low))
115         historyStock$Close <- as.numeric(gsub(',', replacement = '', historyStock$Close))
116         historyStock$Number <- as.numeric(gsub(',', replacement = '', historyStock$Number))
117         historyStock$Value <- as.numeric(gsub(',', replacement = '', historyStock$Value))
118
119         print(paste0(msg, ", Stock information, download complete."))
120     } else{
121         print(paste0(msg, ", No Data found."))
122     }
123     # ============================================================================
124     return(historyStock)
```

```
Console C:/RDATA/
> as.numeric("1,234,567,890")
[1] NA
Warning message:
NAs introduced by coercion
> as.numeric(gsub(',', replacement = '', "1,234,567,890"))
[1] 1234567890
>
```

字串含千分號, 無法轉為數值

濾掉千分號, 字串轉為數值

驗測結果如下圖，原來用 getSymbols 抓取為斷層的部份，改用 getTWSE 可正常取得資料。



完整程式，請至 GitHub 下載

GitHub: https://github.com/dong945/R