自己寫個 Function 抓證券櫃檯買中心的上櫃股票資訊

股票市場一般有上市、上櫃股票，兩者的差別簡述如下：

【上市】

上市股票，是指經證券交易所批准後公開發行，並在集中市場中撮合交易、掛牌買賣的股票。一家公司要在台灣上市，必須符合證交所規定條件，包含設立年限、資本額，及獲利能力等。

【上櫃】

上櫃股票，是指已經公開發行，並於店頭市場以開掛牌買賣的股票。

一家公司要在台灣上櫃，必須符合財團法人櫃檯買賣中心(簡稱 OTC)規定之條件。和上市相比，上櫃所需具備的條件較為寬鬆，其所規定之公司的設立年限、實收資本額和獲利門檻都較低。

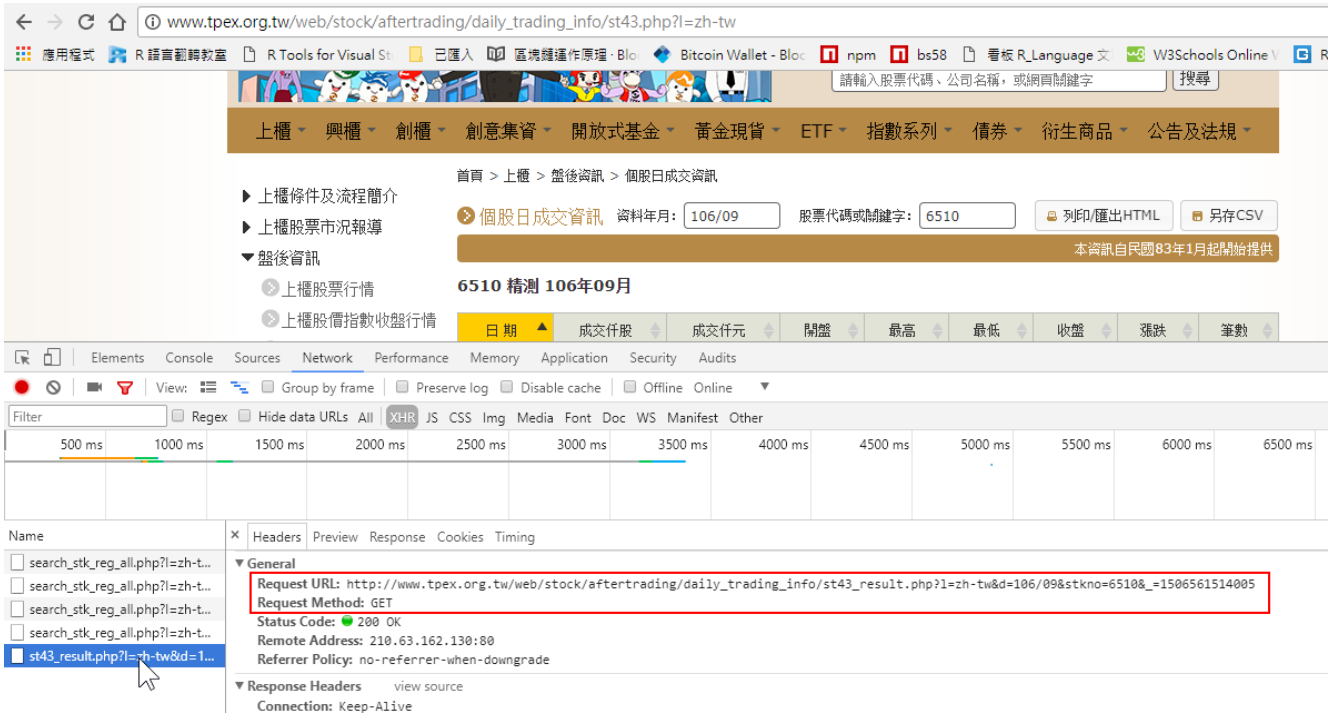
股票上櫃與上市申請條件比較表

項目	上櫃	上市
設立年限	依公司法設立登記滿二個完整會計年度。	依公司法設立登記屆滿三年以上。
實收資本額	新台幣五千萬元以上。	新台幣六億元以上。
獲利能力	其財務報告之稅前淨利占股本之比率最近年度達百分之四以上，且其最近一會計年度決算無累積虧損者；或最近二年度均達百分之三以上者；或最近二年度平均達百分之三以上，且最近一年度之獲利能力較前一年度為佳者。前述財務報告之獲利能力不包含非控制權益之淨利（損）對其之影響。但前揭之稅前淨利，於最近一會計年度不得低於新台幣四百萬元。	其財務報告之稅前淨利符合下列標準之一，且最近一個會計年度決算無累積虧損者。（一）稅前淨利占年度決算之財務報告所列示股本比率，最近二個會計年度均達百分之六以上者。（二）稅前淨利占年度決算之財務報告所列示股本比率，最近二個會計年度平均達百分之六以上，且最近一個會計年度之獲利能力較前一會計年度為佳者。（三）稅前淨利占年度決算之財務報告所列示股本比率，最近五個會計年度均達百分之三以上者。
股權分散	公司內部人及該等內部人持股逾百分之五十之法人以外之記名股東人數不少於三百人，且其所持股份總額合計占發行股份總額百分之二十以上或逾一千萬股。	記名股東人數在一千人以上，公司內部人及該等內部人持股逾百分之五十之法人以外之記名股東人數不少於五百人，且其所持股份合計占發行股份總額百分之二十以上或滿一千萬股者。

科普完上述知識後，繼上次那篇「自己寫個 Function 抓證交所的股票資訊」後，再來一篇抓櫃買中心的上櫃股票。

程序如同上一篇，先至【證券櫃檯買中心】人工走一次下載流程，觀察整個作業程序。
點選[上櫃] -> [盤後資訊] -> [個股日成交資訊]，如下圖

按 F12 進入 Chrome 的開發人員工具，錄制查詢動作，下圖為查 6510



將截取的網址貼至 Chrome，觀察回傳的資料發現是 JSON 格式，如此一來上一篇「自己寫個 Function 抓證交所的股票資訊」的方法只要略為修改，就可以抓取櫃買中心的上櫃股票資料。

觀查網址可發現使用 3 個參數

http://www.tpex.org.tw/web/stock/aftertrading/daily_trading_info/st43_result.php?l=zh-tw&d=106/09&stkno=6510&_=1506562485799

查詢日期：d=106/09，採民國年/月

股票代號：stkno=6510

時間戳記：_=1506562485799，變動值

回傳的 JSON File 中，aaData 即是我們要抓取的 Data，欄位排序如下圖

{"stkNo":"6510","stkName":"\u7cbe\u6e2c","showListPriceNote":false,"showListPriceLink":fa
[["106\/09\/01","228","323,060","1,420.00","1,435.00","1,405.00","1,405.00","-15.00","245"],
["106\/09\/04","195","274,974","1,405.00","1,425.00","1,395.00","1,405.00","0.00","211"],
["106\/09\/05","179","253,783","1,420.00","1,430.00","1,410.00","1,425.00","20.00","196"],
["106\/09\/06","249","353,044","1,420.00","1,435.00","1,395.00","1,405.00","-20.00","257"],
["106\/09\/07","184","260,171","1,420.00","1,430.00","1,410.00","1,410.00","5.00","177"],

個股日成交資訊　資料年月：106/09　股票代碼或關鍵字：6510　列印/匯出HTML　另存CSV

本資訊自民國83年1月起開始提供

6510 精測 106年09月

日 期	成交仟股	成交仟元	開盤	最高	最低	收盤	漲跌	筆數
106/09/01	228	323,060	1,420.00	1,435.00	1,405.00	1,405.00	-15.00	245
106/09/04	195	274,974	1,405.00	1,425.00	1,395.00	1,405.00	0.00	211
106/09/05	179	253,783	1,420.00	1,430.00	1,410.00	1,425.00	20.00	196
106/09/06	249	353,044	1,420.00	1,435.00	1,395.00	1,405.00	-20.00	257
106/09/07	184	260,171	1,420.00	1,430.00	1,410.00	1,410.00	5.00	177
106/09/08	228	324,821	1,415.00	1,445.00	1,415.00	1,415.00	5.00	247

函數：getGreTai(stkno, from = c(year, month), to = c(year, month))，預設有個參數，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。

Stkno = “6510”，預設為中華精測(6510)

```
26 # ============================================================================================================
27 getGreTai <- function(stkno = "6510",
28                        from = c(as.integer(format(Sys.Date(),"%Y")), as.integer(format(Sys.Date(),"%m"))),
29                        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 42~49，主要是檢查 RStudio 是否已載入 jsonlite、xts 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

```
39     # ============================================================================================================
40     # Using this function, you must library(jsonlite)、library(xts)
41     # ============================================================================================================
42     packages <- gsub("package:", replacement = "", search())
43     if (!("jsonlite" %in% packages)){
44         print("Error: The 'jsonlite' package has not been loaded.")
45         return(NULL)
46     } else if (!("xts" %in% packages)){
47         print("Error: The 'xts' package has not been loaded.")
48         return(NULL)
49     }
```

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 53~59，檢查年度值，櫃買中心的資訊自民國 83 年 1 月起開始提供.
行 60~71，判斷 from 及 to 的內容，若僅有年度，則補上月份.
行 72~73，將年的部份由西元年轉為民國年.

```
50    # ============================================================================================
51    # parameter check & parse
52    # ============================================================================================
53    if (from[1] > to[1]){
54        print("The starting year is greater than the deadline.")
55        return(NULL)
56    } else if (from[1] < 1994 | to[1] < 1994){
57        print("This information has been available since January 1994.")
58        return(NULL)
59    }
60    nowYear <- as.integer(format(Sys.Date(),"%Y"))
61    nowMonth <- as.integer(format(Sys.Date(),"%m"))
62    MM <- c("01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12")
63    queryDate <- NULL
64    historyStock <- NULL
65    startYM <- from
66    endYM <- to
67    if (length(startYM) == 1) startYM <- c(startYM, 1)
68    if (length(endYM) == 1){
69        ifelse (endYM[1] == nowYear,
70                endYM <- c(endYM, nowMonth), endYM <- c(endYM, 12))
71    }
72    startYM[1] <- startYM[1]-1911
73    endYM[1] <- endYM[1]-1911
```

年的部份轉成民國年

行 89~101，計算出查詢區間的明細，待會組 URL 時，參數 d 的內容，格式為 yyy/mm

```
88    # ============================================================================================
89    if (startYM[1] == endYM[1]) {
90        queryDate <- paste(startYM[1], MM[startYM[2]:endYM[2]], sep = "/")
91    } else if ((endYM[1] - startYM[1]) == 1) {
92        queryDate <- c(paste(startYM[1], MM[startYM[2]:12], sep = "/"),
93                       paste(endYM[1], MM[1:endYM[2]], sep = "/"))
94    } else {
95        tmpY <- c((startYM[1]+1):(endYM[1]-1))
96        queryDate <- paste(startYM[1], MM[startYM[2]:12], sep = "/")
97        for (tY in tmpY){
98            queryDate <- c(queryDate, paste(tY, MM, sep = "/"))
99        }
100       queryDate <- c(queryDate, paste(endYM[1], MM[1:endYM[2]], sep = "/"))
101   }
```

計算區間
格式為yyy/mm

行 1075，組查詢網址，傳入 3 個參數；行 108，真正到櫃買中心抓資料，判斷回傳的 JSON，取 length(jsonData$aaData) != 0，組 OHLC 格式(Open, High, Low, Close)

```
102   # ============================================================================================
103   # to GreTai get History Stock
104   # ============================================================================================
105   for (qyDate in queryDate){
106       ttime <- as.character(as.integer(as.POSIXct(Sys.time()))*100)
107       gretaiUrl <- paste0(url, "&d=", qyDate, "&stkno=", stkno, "&_=", ttime)
108       jsonData <- fromJSON(gretaiUrl, flatten = TRUE)
109       if (length(jsonData$aaData) != 0){
110           tmpStock <- data.frame(jsonData$aaData[, 1],
111                                  jsonData$aaData[, 4:7],
112                                  jsonData$aaData[, 3],
113                                  jsonData$aaData[, 2],
114                                  stringsAsFactors = FALSE)
115           historyStock <- rbind(historyStock, tmpStock)
116       }
117
118   }
```

組櫃買中心網址

抓資料

判斷是否有資料

行 129，設定 column name，除了 OHLC 之外，若要 quantmod 套件的 chartSeries 畫出正確的圖形，column name 中需要有一個”Volume”。
行 130~136，資料格式轉換，行 130，利用自訂函數 CNV_DATE()，透過 sapply 函數將民國年轉為西元年，行 131~136，則是將 string 轉為 number，因為含千分號(,)，故先要濾掉。
行 135, 136，因為單位分別為仟元及仟股，故乘以 1000，轉為元及股。
行 138，將 data frame 轉為 xts 格式

```
119   # ================================================================================
120   # Convert Data Format:
121   # If the string has a comma, it can not be converted to a value.
122   # gsub(',', replacement = '', x) --> remove comma
123   #
124   # Ex. as.numeric("196,857,432") --> Warning message: NAs introduced by coercion
125   #     as.numeric("196857432") --> 196857432
126   # ================================================================================
127   if (!is.null(historyStock)){                                    column name
128       # colnames(historyStock) <- c("Date", "Open", "High", "Low", "Close", "Number", "Value")
129       colnames(historyStock) <- c("Date", "Open", "High", "Low", "Close", "Volume", "Number")
130       historyStock$Date <- sapply(historyStock$Date, CNV_DATE)
131       historyStock$Open <- as.numeric(gsub(',', replacement = '', historyStock$Open))
132       historyStock$High <- as.numeric(gsub(',', replacement = '', historyStock$High))
133       historyStock$Low <- as.numeric(gsub(',', replacement = '', historyStock$Low))
134       historyStock$Close <- as.numeric(gsub(',', replacement = '', historyStock$Close))
135       historyStock$Volume <- as.numeric(gsub(',', replacement = '', historyStock$Volume))*1000
136       historyStock$Number <- as.numeric(gsub(',', replacement = '', historyStock$Number))*1000
137       # data frame to xts
138       historyStock <- xts(historyStock[, -1], order.by = as.Date(historyStock[, 1]))
139
140       print(paste0(msg, ", Stock information, download complete. rows=", nrow(historyStock)))
141   } else{
142       print(paste0(msg, ", No Data found."))
143   }
```

```
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

完整程式，請至 GitHub 下載
GitHub: https://github.com/dong945/R

完整程式碼

```
# ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
# Author: TUNG-SHENG, CHEN
# DATE:      2017/09/27
# Package: jsonlite, xts
# ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
# install.packages(c("jsonlite", "xts"))
library(jsonlite)
library(xts)
# ==============================================================================================
# Parameter:
#      stockNo: String, Stock code of TWSE, default: 6510 (CHPT)
#      from:        Vector, c(year, month), default: now
#      to:          Vector, c(year, month), default: now
# Examples:
#      ## return '6510', this month
#      getGreTai()
#
#      ## return '6510', from 2016/01 ~ now
#      getGreTai("6510", 2016)
#
#      ## return '6510', from 2015/01 ~ 2016/12
#      getGreTai("6510", 2015, 2016)
#
#      ## return '6510', from 2015/06 ~ 2017/03
#      getGreTai("6510", c(2105, 6), c(2017, 3))
# ==============================================================================================
```

```
getGreTai <- function(stkno = "6510",
                      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")))) {
  # ============================================================================================
  # Inside Function: Change Date to yyyy-mm-dd
  # ============================================================================================
  CNV_DATE <- function(x){
    TMP <- strsplit(x, split = "/")
    paste(as.integer(TMP[[1]][1])+1911, TMP[[1]][2], TMP[[1]][3], sep = "-")
  }
  # ============================================================================================
  url <- "http://www.tpex.org.tw/web/stock/aftertrading/daily_trading_info/st43_result.php?l=zh-tw"
  # ============================================================================================
  # Using this function, you must library(jsonlite)、library(xts)
  # ============================================================================================
  packages <- gsub("package:", replacement = "", search())
  if (!("jsonlite" %in% packages)){
    print("Error: The 'jsonlite' package has not been loaded.")
    return(NULL)
  } else if (!("xts" %in% packages)){
    print("Error: The 'xts' package has not been loaded.")
    return(NULL)
  }
  # ============================================================================================
  # parameter check & parse
  # ============================================================================================
  if (from[1] > to[1]){
```

```
        print("The starting year is greater than the deadline.")
        return(NULL)
} else if (from[1] < 1994 | to[1] < 1994){
        print("This information has been available since January 1994.")
        return(NULL)
}
nowYear <- as.integer(format(Sys.Date(),"%Y"))
nowMonth <- as.integer(format(Sys.Date(),"%m"))
MM <- c("01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12")
queryDate <- NULL
historyStock <- NULL
startYM <- from
endYM <- to
if (length(startYM) == 1) startYM <- c(startYM, 1)
if (length(endYM) == 1){
        ifelse (endYM[1] == nowYear,
                endYM <- c(endYM, nowMonth), endYM <- c(endYM, 12))
}
startYM[1] <- startYM[1]-1911
endYM[1] <- endYM[1]-1911
# Check
if (startYM[2] < 1 | startYM[2] > 12 | endYM[2] < 1 | endYM[2] > 12){
        print("Month must be between 1 ~ 12.")
        return(NULL)
} else if (startYM[1] > endYM[1]) {
    print("Start year is greater than the End year.")
    return(NULL)
```

```
} else if ((startYM[1] == endYM[1]) & (startYM[2] > endYM[2])) {
   print("Start month is greater than the End month.")
   return(NULL)
}
msg <- paste0("Stock Code=", stkno,
                        ", from(", (startYM[1]+1911), ", ", startYM[2], ")",
                        " to(", (endYM[1]+1911), ", ", endYM[2], ")")
# ================================================================================
if (startYM[1] == endYM[1]) {
      queryDate <- paste(startYM[1], MM[startYM[2]:endYM[2]], sep = "/")
} else if ((endYM[1] - startYM[1]) == 1) {
      queryDate <- c(paste(startYM[1], MM[startYM[2]:12], sep = "/"),
                           paste(endYM[1], MM[1:endYM[2]], sep = "/"))
} else {
      tmpY <- c((startYM[1]+1):(endYM[1]-1))
      queryDate <- paste(startYM[1], MM[startYM[2]:12], sep = "/")
      for (tY in tmpY){
            queryDate <- c(queryDate, paste(tY, MM, sep = "/"))
      }
      queryDate <- c(queryDate, paste(endYM[1], MM[1:endYM[2]], sep = "/"))
}
# ================================================================================
# to GreTai get History Stock
# ================================================================================
for (qyDate in queryDate){
      ttime <- as.character(as.integer(as.POSIXct(Sys.time()))*100)
      gretaiUrl <- paste0(url, "&d=", qyDate, "&stkno=", stkno, "&_=", ttime)
```

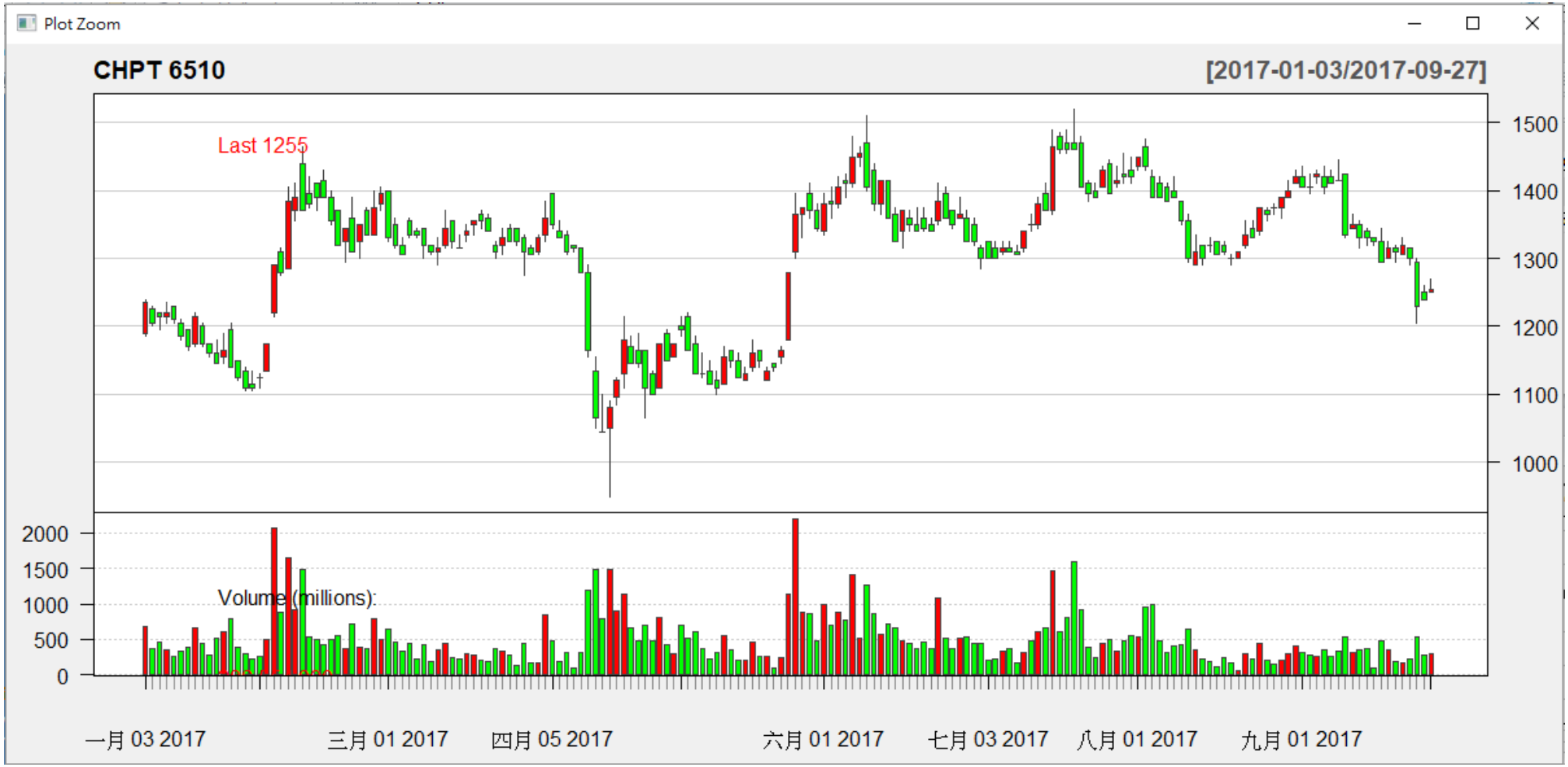
```
        jsonData <- fromJSON(gretaiUrl, flatten = TRUE)
        if (length(jsonData$aaData) != 0){
                tmpStock <- data.frame(jsonData$aaData[, 1],
                                                jsonData$aaData[, 4:7],
                                                jsonData$aaData[, 3],
                                                jsonData$aaData[, 2],
                                                stringsAsFactors = FALSE)
                historyStock <- rbind(historyStock, tmpStock)
        }

}
# ================================================================================
# Convert Data Format:
# If the string has a comma, it can not be converted to a value.
# gsub(',', replacement = '', x) --> remove comma
#
# Ex. as.numeric("196,857,432") --> Warning message: NAs introduced by coercion
#         as.numeric("196857432") --> 196857432
# ================================================================================
if (!is.null(historyStock)){
        # colnames(historyStock) <- c("Date", "Open", "High", "Low", "Close", "Number", "Value")
        colnames(historyStock) <- c("Date", "Open", "High", "Low", "Close", "Volume", "Number")
        historyStock$Date <- sapply(historyStock$Date, CNV_DATE)
        historyStock$Open <- as.numeric(gsub(',', replacement = '', historyStock$Open))
        historyStock$High <- as.numeric(gsub(',', replacement = '', historyStock$High))
        historyStock$Low <- as.numeric(gsub(',', replacement = '', historyStock$Low))
        historyStock$Close <- as.numeric(gsub(',', replacement = '', historyStock$Close))
```

```
        historyStock$Volume <- as.numeric(gsub(',', replacement = '', historyStock$Volume))*1000
        historyStock$Number <- as.numeric(gsub(',', replacement = '', historyStock$Number))*1000
        # data frame to xts
        historyStock <- xts(historyStock[, -1], order.by = as.Date(historyStock[, 1]))

        print(paste0(msg, '', Stock information, download complete. rows=", nrow(historyStock)))
    } else{
        print(paste0(msg, '', No Data found."))
    }
    # ==========================================================================================
    return(historyStock)
}

# ==========================================================================================
# TEST
# ==========================================================================================
library(quantmod)
tw6510 <- getGreTai("6510", c(2017, 1), c(2017, 9))
chartSeries(tw6510, theme = chartTheme("white", up.col = "red", dn.col = "green"),
                    name = "CHPT 6510", show.grid = TRUE)
```


Plot Zoom
CHPT 6510
[2017-01-03/2017-09-27]
Last 1255
1500
1400
1300
1200
1100
1000
2000
1500
1000
500
0
Volume (millions):
一月 03 2017
三月 01 2017
四月 05 2017
六月 01 2017
七月 03 2017
八月 01 2017
九月 01 2017