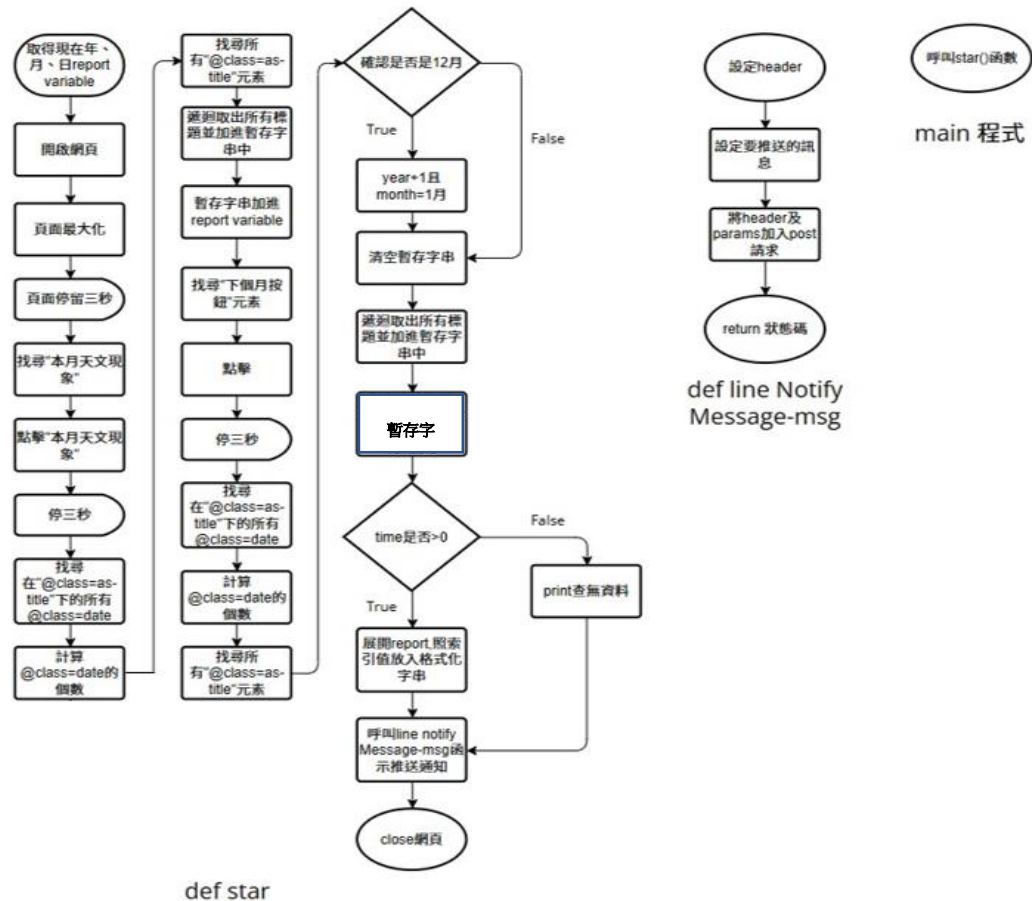


## 天文現象推播

1. 簡介
2. 透過 Python seleium 及 Chromedriver 爬取中央氣象局本月及下個月的天文現象，再藉由 LINE Notify 傳送至 LINE 聊天室
3. 流程圖



- #### 4. 程式說明
- 引用函式庫
  - 定義當前時刻
  - 設定爬取網頁為「交通部中央氣象局-每月星象」
  - 設定 LINE Notify 使用者權杖

```
from selenium import webdriver
import time
import requests
import datetime
tonow = datetime.datetime.now()
report = []

driverPath=r"C:\Users\acer\Downloads\chromedriver-win64\chromedriver.exe"
browser=webdriver.Chrome(driverPath)
```

- report 字串存放要顯示在 LINE 訊息中的資料
- 使用 find\_element('xpath')定位到"本月天文天文現象"欄

```
def star():
    reformat = ''
    year=tonow.year
    month=str(tonow.month)
    report.append(year)
    report.append(month)

    browser.get(url)
    browser.maximize_window()
    time.sleep(3)
    button=browser.find_element('xpath','//*[@id="calendar"]/div[1]/ul/li[2]/a')
    button.click()
    time.sleep(3)
```

爬取當月天文現象及日期，放進暫存字串

```
#本月天文現象
tits=browser.find_elements('xpath','//*[@class="as-title"]//*[@class="date"]')
times=0
for i in tits:
    times+=1
tits=browser.find_elements('xpath','//*[@class="as-title"]')

num=1

for i in tits:
    if times<=0:
        break
    #把標題加進暫存字串中
    reformat += "\n\n"
    reformat += i.text

    print(num)
    print(i.text)
    print("\n")

    times-=1
    num+=1
report.append(reformat)
```

若本月為 12 月則須對下個月之時間資訊另做處理

```
#計算月份與進位
month = (tonow.month+1)%12
if(((tonow.month+1)/12)==1):
    year += 1
report.append(year)
report.append(month)
```

- 爬取下個月的資訊
- 清空 `reformat` 字串儲存下一個月的天文現象資訊。如果該月無資料，則列印 "查無資料"，並將它添加到 `report` 中。

```
#下個月天文現象
button=browser.find_element('xpath','/html/body/div[3]/main/div/div/div/div[1]/div[1]/div[2]/div/button[2]')
button.click()
time.sleep(3)

tits=browser.find_elements('xpath','//*[@class="as-title"]//*[@class="date"]')
times=0
for i in tits:
    times+=1
tits=browser.find_elements('xpath','//*[@class="as-title"]')
```

```
#清空暫存字串
reformat = ''
num=1
if times>0:
    for i in tits:
        if times<=0:
            break
        reformat += "\n\n"
        reformat += i.text
        print(num)
        print(i.text)
        print("\n")
        times-=1
        num+=1
    report.append(reformat)
else:
    print("查無資料")
```

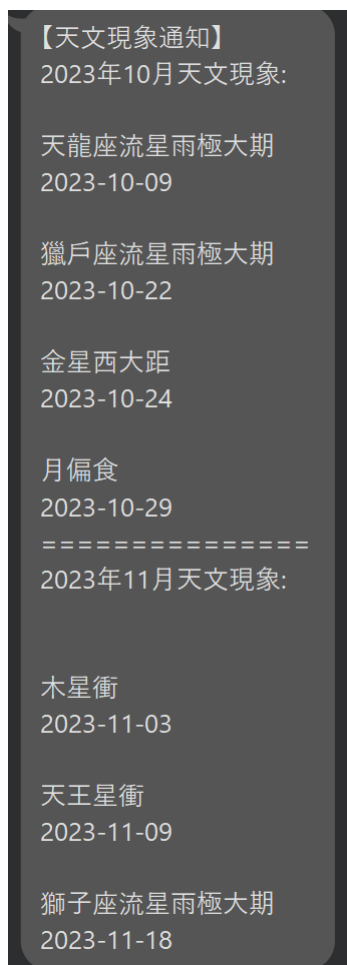
- 格式化 `report` 列表中的資訊
- 呼叫 `lineNotifyMessage_msg` 函數將格式化後的資訊通知給 LINE 帳號。
- 爬取所需訊息後關閉瀏覽器。

```
#展開report，照索引值依序放入{}
reformat = "\n{}年{}月天文現象:{}\n===== \n{}年{}月天文現象:\n{}".format(*report)
lineNotifyMessage_msg(token,reformat)
browser.close()
```

- 設定 token 及 message 等訊息，使用 LINE Notify 服務發送通知。
- 執行 star 函式

```
def lineNotifyMessage_msg(token,msg):  
    headers={  
        "Authorization":"Bearer "+ token,  
        #"Content-Type":"application/x-www-form-urlencoded"  
    }  
    payload={'message':msg}  
    r=requests.post("https://notify-api.line.me/api/notify",headers=headers,params=payload)  
    #print(r.status_code)  
    return r.status_code  
  
star()
```

## 5. 運行結果



## 6. 結論

本作品使用 Python，結合 Selenium 與 LINE Notify，並可設定工作排程器，讓爬蟲程式每月觸發，自動擷取星象與日期並通知使用者，不再需要繁瑣的查詢步驟，對於天文愛好者是非常方便的工具。