

報告者:許知穎

指導老師:簡志峰老師

# 目錄 CONTENTS



選取資料動機
Data selection and motivation



問題與解決 Problems and solutions



爬蟲目的說明 Purpose

- (1)遊戲資料搜索
- (2)拓元售票系統購票

**0**4

總結與感想 Conclusions



# 選取資料動機



Data selection and motivation



## 01 選取資料動機 & 資料來源

1. 遊戲資料搜索

資料讀取/存取檔案: armors\_jp.csv

網頁: https://w.atwiki.jp/rockyou11

2. 拓元售票系統搶票

網頁: https://tixcraft.com/





# 爬蟲目的說明



Purpose



### 2-1 遊戲資料搜索

#### <u>需求</u>

- 1. 將檔案中第二欄的遊戲角色名稱, 丟入網頁 <u>https://w.atwiki.jp/rockyou11</u>中, 進行搜索。
- 2. 將搜索到的資料連結輸出回原檔案中, 需以「,」做為分隔。



#### 檔案原始內容



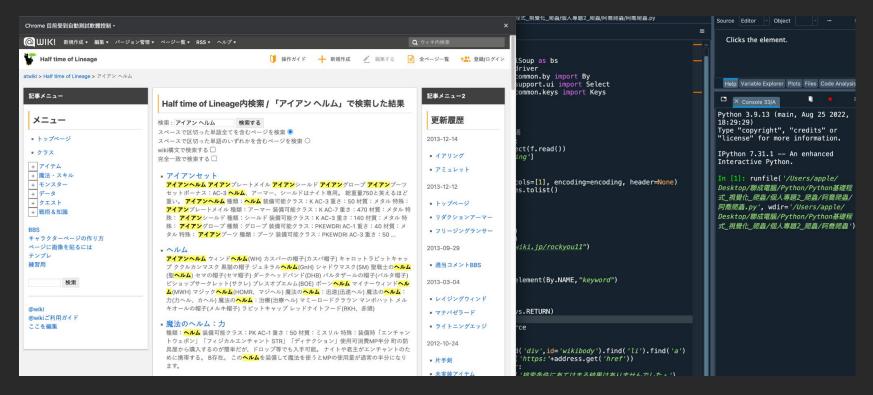
#### 完成內容

⊗ ⊘ armors_jp.csv		
20001	レザー キャップ	検索条件にあてはまる結果はありませんでした。
20002	レザーヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%AC%E3%82%B6%E3%83%BC%20%E3%8
20003	アイアン ヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%82%A2%E3%82%A4%E3%82%A2%E3%83%8
20004	シャドウ マスク	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%82%B7%E3%83%A3%E3%83%89%E3%82%
20005	ナイトフード	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%8A%E3%82%A4%E3%83%88%E3%83%8
20006	ナイト バイザー	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%8A%E3%82%A4%E3%83%88%20%E3%
20007	ドワヴィッシュ アイアン ヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%89%E3%83%AF%E3%83%B4%E3%82%A
20008	マイナー ウィンドヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%9E%E3%82%A4%E3%83%8A%E3%83%8
20009	デーモン ヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%87%E3%83%BC%E3%83%A2%E3%83%8
20010	デスナイト ヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%87%E3%82%B9%E3%83%8A%E3%82%
20011	マジック ヘルム	https://w.atwiki.jp/rockyou11/7cmd=word&word=%E3%83%9E%E3%82%B8%E3%83%83%E3%82%A
20012	ウィザードの帽子	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%82%A6%E3%82%A3%E3%82%B6%E3%83%8
20013	魔法のヘルム:迅速	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E9%AD%94%E6%B3%95%E3%81%AE%E3%83%
20014	魔法のヘルム:治癒	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E9%AD%94%E6%B3%95%E3%81%AE%E3%83%
20015	魔法のヘルム:力	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E9%AD%94%E6%B3%95%E3%81%AE%E3%83%
20016	マンポ ハット	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%9E%E3%83%B3%E3%83%9C%20%E3%8
20017	マミーロード クラウン	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%9E%E3%83%9F%E3%83%BC%E3%83%A
20018	メルキオールの帽子	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%A1%E3%83%AB%E3%82%AD%E3%82%
20019	クラウン	https://w.atwiki.jp/rockyou11/7cmd=word&word=%E3%82%AF%E3%83%A9%E3%82%A6%E3%83%8
20020	ジェネラル ヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%82%B8%E3%82%A7%E3%83%8D%E3%83%
20021	機敏のヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E6%A9%9F%E6%95%8F%E3%81%AE%E3%83%9
20022	パランカ ヘルム	https://w.atwiki.jp/rockyou11/7cmd=word&word=%E3%83%90%E3%83%A9%E3%83%B3%E3%82%
20023	ウィンドヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%82%A6%E3%82%A3%E3%83%B3%E3%88%
20024	反王のヘルム	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E5%8F%8D%E7%8E%8B%E3%81%AE%E3%83%
20025	パルタザールの帽子	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%90%E3%83%AB%E3%82%BF%E3%82%E
20026	ナイト ビジョン	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%8A%E3%82%A4%E3%83%88%20%E3%8
20027	レッドナイトフード	https://w.atwiki.jp/rockyou11/?cmd=word&word=%E3%83%AC%E3%83%83%E3%83%89%E3%83%8

### 2-1 遊戲資料搜索



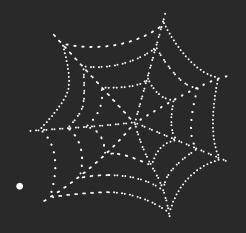
#### 程式執行畫面



#### \_\_\_\_

#### 需求

- 1. 能提前登入會員帳號 (拓元售票系統規定 需有登入且認證過才可購票。
- 2. 能於指定時間開始購票。
- 3. 可選擇購票場次/票價區域/張數。
- 4. 可自動跳過無法點選的選項 (已售完)。



#### 購票網頁內容

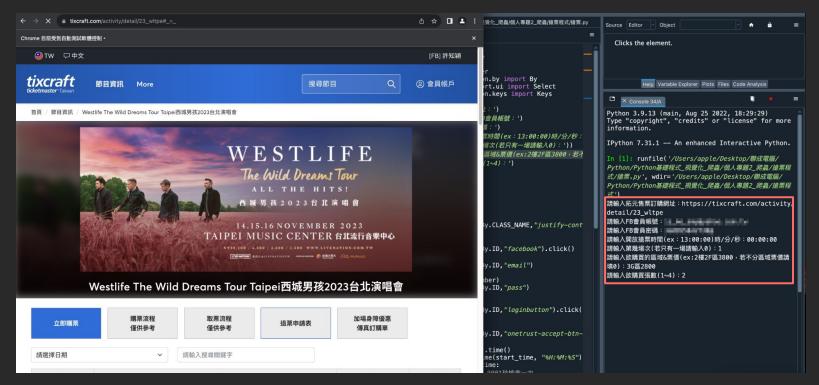






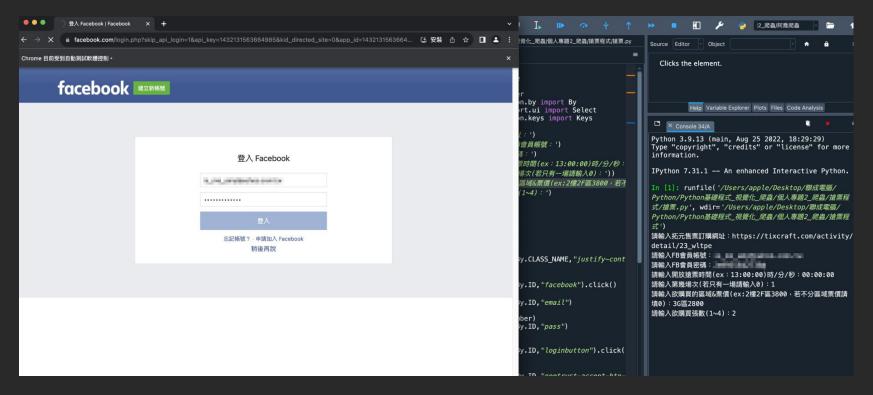


程式執行畫面:輸入購票網頁,輸入相關需求資訊





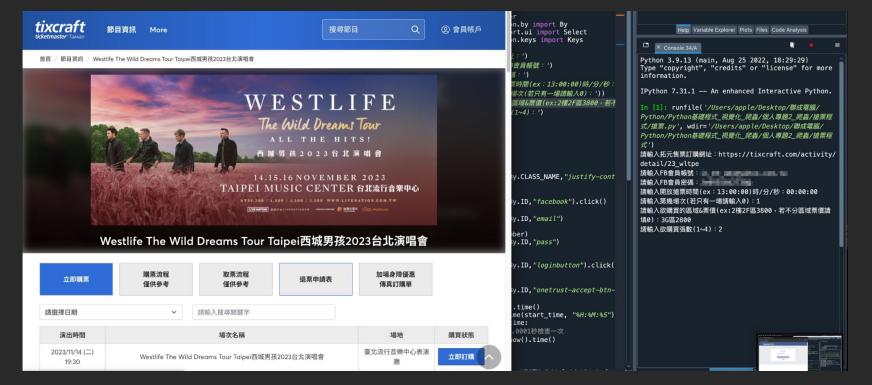
#### 程式執行畫面:會員登入





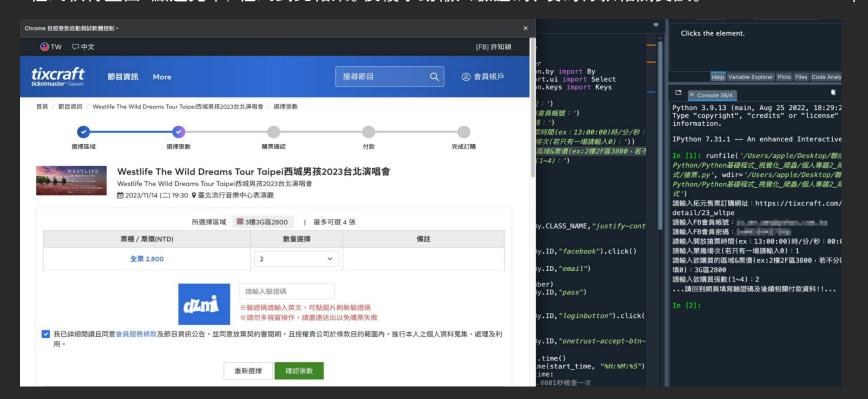


程式執行畫面:進行購票(點選訂購、場次、票價區間)





程式執行書面:點選完畢,程式到此結束。後續手動輸入驗證碼、填寫付款相關資訊。





# 問題與解決



Problems and solutions



### 3-1 遊戲資料搜索

#### 問題與解決

問題一:日文無法用utf8編碼。

解答一:用chardet套件自動檢測編碼。

問題二: 抓取href網址。

解答二:使用get('href')抓到連結網址。

問題三: 有些檔案搜尋不到資料, 無法抓取網址出現錯誤。

解答三: 使用try except,讓搜尋不到資料的填入找不到結果。

問題四:欄位本身沒有索引值,若新增一欄會將新增那一欄的

第一筆資料視為索引值,導致數量不一錯誤。

解答四: 指定header=None,將所有行當作資料,不將第一行

**• 机**為索引,寫入時亦同。

```
# 使用chardet套件自動檢測編碼
with open(fn, 'rb') as f:
    result = chardet.detect(f.read())
encoding = result['encoding']
# 指定header=None,表示將所有行都當作資料,不將第一行視為索引
df = pd.read_csv(fn, usecols=[1], encoding=encoding, header=None
search page = df[1].values.tolist()
# 建立list存放網址
addresses=[]
driver=webdriver.Chrome()
driver.implicitly wait(3)
driver.get("https://w.atwiki.jp/rockyou11")
# 輸入期別進行查詢
for i in search page:
   element=driver.find_element(By.NAME, "keyword")
    element.click()
    element.clear()
    element.send keys(i)
   element.send_keys(Keys.RETURN)
   html=driver.page_source
   soup=bs(html,"lxml")
       address=soup.find('div',id='wikibody').find('li').find('a'
       addresses.append('https:'+address.get('href')
    except AttributeError:
       addresses.append('検索条件にあてはまる結果はありませんでした。')
driver.quit()
# 儲存回CSV檔案,指定header=False,表示不將第一行當作列索引寫入
df = pd.read csv(fn.sep=',', encoding=encoding, header=None)
df.insert(2, '新欄位', addresses)
df.to_csv('armors_jp.csv', index=False, encoding=encoding, header=False)
```

web=input('請輸入拓元售票訂購網址:')
account\_number=input('請輸入FB會員帳號:')
password=input('請輸入FB會員經碼:')
start\_time=input('請輸入開放搶票時間(ex:13:00:00)時/分/秒:')
sessions=str(input('請輸入第幾場次(若只有一場請輸入0):'))
price=str(input('請輸入欲購買的區域必票價(ex:2樓2F區3800,若不分區域票價請填0):')
tickets=input('請輸入欲購買張數(1~4):')

#### 問題與解決

問題一:想做到提前會員登入,等到特定時間再進行購票。

解答一:用date time抓現在時間及設定目標時間,使現在時間=目標時間時才開始執行後半段程式碼。

問題二:有些演唱會僅一場次,有些是多場次。下方的立即訂購無法定位。

解答二:觀察原始碼發現,若僅一場次則r後無[],若有多場次, tr後[]內

則是場次編號,以此來判斷定位的路徑名稱。

問題三:即便定位確認無誤, selenium還是點不到按鍵。

解答三:跳過selenium透過執行JavaScript來點擊元素。

#### 問題四:

有些場次均一票價,無須選擇票價區域。

網頁原始碼有些區域票價的文字放在li>有些放在<a>。

已售完的區域票價文字會放在<font>。

解答四:先用if not elements條件判斷來檢查elements是否為空或False, 確定是<a>還是。

再使用for迴圈+try except來嘗試點選至可以點選為止。

```
# 等時間到了再執行
current_time = datetime.now().time()
target_time = datetime.strptime(start_time, "%H:%M:%S").time()
while current_time < target_time:
    time.sleep(0.0001) # 每0.0001秒檢查一次
    current_time = datetime.now().time()
```

```
# 立即訂購(下):tr[第幾場(從1開始)],若只有一場就沒有索引
if sessions=='0':
    session='//*[@id=\"gameList\"]/table/tbody/tr/td[4]/button'
else:
    session='//*[@id=\"gameList\"]/table/tbody/tr['+sessions+']/td[4]/button'
```

```
driver.implicitly_wait(2)
# 立即訂購(上)-各場次都一樣
element=driver.find_element(By.XPATH,"//*[@id=\"tab-func\"]/li[1]/a/div")
# seleniun無法貼可能是元素重量或隱藏,透過執行JavaScript來點擊元素。
# arguments[0] 代表傳遞給 execute_script() 方法的第一個參數,即我們要點擊的 element 物件。等於click()
driver.execute_script("arguments[0].click();", element)
```

```
# 選區域票價:部份票價在 * 部份在<a>。部份是均一票價不用選擇,就需直接跳過此段。
# 先尋找所有符合條件的 <a> 元素和  元素
if price != '0':
    elements = driver.find_elements(By.XPATH, "//a[contains(.,'"+price+"')]")
    if not elements:
        elements = driver.find_elements(By.XPATH, "//li[contains(.,'"+price+"')]")
    # 找到第一個符合條件的元素,然後點選。預防售完的票價無法點選,導致整個程式需從頭輸入資料重新跑。
    for element in elements:
        try:
            driver.execute_script("arguments[0].click();", element)
            break # 成功找到並點選元素後,跳出迴圈,繼續往下執行
        except:
            pass # 如果點選失敗,繼續尋找下一個符合條件的元素
```

### 3-2 高鐵售票系統購票

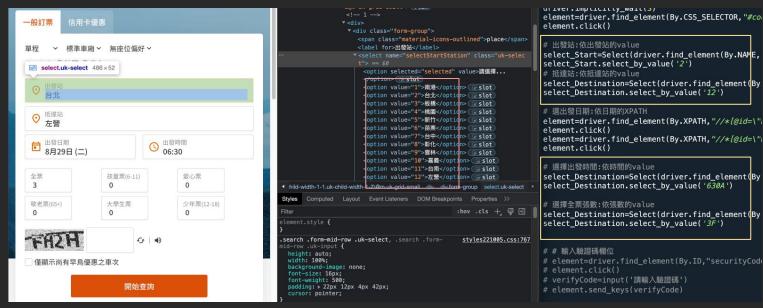
······

#### 問題與解決

問題一: 出發/抵達站、出發時間、張數的下拉式選單. 用 select函數建立物件回傳 值會報錯。

解答一: 仔細觀察原始碼,發現他們都有另外定義 value, 需回傳相對應的 value才可以成功點選。

問題二: 資料全部輸入完成後, 按開始 查詢, 就會當掉。---尚未解決...







# 總結與感想



Conclusions



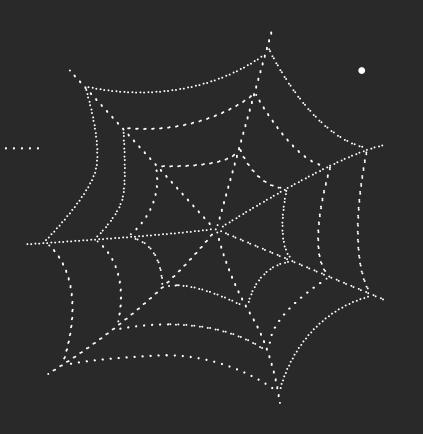
### 04總結與感想

·\*\*

爬蟲網站真的很多不同的防爬機制, 必須針對各個機 制做出相對應的指令才能順利執行。

寫程式碼之前, 觀察網頁原始碼越透徹, 越能順利寫出爬蟲程式。

訂定主題練習時,針對自己有興趣的方面去找,會更有動力,寫完之後也會更有成就感。





報告結束,感謝聆聽

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon** and infographics & images by **Freepik**