自動填入帳號及密碼，並登入facebook

透過Chromdriver，並前往facebook 登入頁面。可設定要前往的網址，並利用driver.get（url）自動前往該畫面，如下程式碼。

In [1]:

```
from selenium.webdriver.common.keys import Keys
from selenium.webdriver import DesiredCapabilities
from selenium.common.exceptions import NoSuchElementException
from selenium import webdriver
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from bs4 import BeautifulSoup as Soup
from selenium.webdriver.common.action_chains import ActionChains
from pandas.core.frame import DataFrame
from bs4 import BeautifulSoup as Soup
import time
import pandas as pd
import json
```

In [2]:

```
# ------ 設定前往臉書 ------
url = 'https://www.facebook.com/'

# ------ 透過Browser Driver 開啟 Chrome ------
driver = webdriver.Chrome('./chromedriver')

# ------ 前往臉書首頁 ------
driver.get(url)
```

如果要找尋特定位置，通常會使用網頁元素中的id標籤，由上述步驟我們可以發現輸入電子郵件或手機號碼的id=email，輸入密碼的id=pass。找到網頁元素後只需用send_keys這個function就能自動輸入帳號密碼囉，如下程式碼所示!! 是不是很簡單呢~

In [ ]:

```
# ------ 登入的帳號與密碼 ------
#username = ''
#password = ''
username = input('Username: ')
password = input('Passwd: ')

# ------ 輸入賬號密碼 ------
WebDriverWait(driver, 30).until(EC.presence_of_element_located((By.XPATH, '//*[
elem = driver.find_element_by_id("email")
elem.send_keys(username)

elem = driver.find_element_by_id("pass")
elem.send_keys(password)

elem.send_keys(Keys.RETURN)
time.sleep(5)
```

2.前往要爬取的粉絲專頁

登入facebook後，前往BBC中文網粉絲專頁

In [4]:

```
#檢查有沒有被擋下來
if len(driver.find_elements_by_xpath("//*[contains(text(), '你的帳號暫時被鎖住')]"
    driver.find_elements_by_xpath("//*[contains(text(), '是')]")[1].click()

# 切換頁面
spec_url = 'https://www.facebook.com/bbcnewstrad'
driver.get(spec_url)
```

抓取文章框架

***注意事項：**

先使用Javascript 中的window.scrollTo()讓程式自動下拉滾輪，直到所需的貼文都出現!

In [5]:

```
# 建立一個scroll function
# scrolltimes代表頁面滾動的次數

def scroll(scrolltimes):
  for i in range(scrolltimes):
    # 每一次頁面滾動都是滑到網站最下方
    js = 'window.scrollTo(0, document.body.scrollHeight);'
    driver.execute_script(js)
    time.sleep(2)

# 呼叫scroll function，就會直接滾動頁面
scroll(30)
```

使用BeautifulSoup套件，把class name放進 find_all() function裡面，並執行以下程式碼，把全部貼文內容都抓取下來!

***BeautifulSoup補充說明：**

- find_all() 為抓取所有符合此class name的元素內容
- find() 為抓取第一個符合此class name的元素內容

抓取按讚數

使用BeautifulSoup套件，把找到的 span 標籤及class name 'pcp91wgn' 放進 find() 裡面

In [7]:

```
# 抓取所有貼文
soup = Soup(driver.page_source, "lxml")
frames = soup.find_all(class_='du4w35lb k4urcfbm l9j0dhe7 sjgh65i0')
```

連續抓取按讚數

成功抓取第一篇貼文後，接下來我們使用for迴圈，把所有貼文的按讚數儲存在一個list裡面，請參考以下程式碼

In [8]:

```
# 建立一個空的list
likes = []

# 抓取每一篇貼文的按讚數
# class name可能會修改，需要定期偵錯!
for f in frames:
    thumb = f.find('span',class_="pcp91wgn")

    # 有些貼文沒有按讚數，所以抓下來的東西是None，因此直接append 0
    if(thumb == None):
        likes.append('0')
    else:
        likes.append(thumb.text)
```

資料整理

In [9]:

```
for i in range(len(likes)):
    # 處理出現 '\xa0萬' 的數值
    if(likes[i].find('\xa0萬') != -1):
        likes[i] = int(float(likes[i][:likes[i].find('\xa0萬')])*10000)

    # 處理有出現 ',' 的數值
    else:
        likes[i] = int(likes[i].replace(',',''))
```

In [10]:

```
print(likes)
print(len(likes))
```

```
[4187, 245, 175, 72, 354, 17, 216, 74, 269, 419, 637, 328, 1005, 525,
73, 283, 272, 707, 610, 82, 276, 117, 844, 857, 97, 7051, 599, 3008,
90, 46, 203, 108, 146, 993, 973, 402, 1626, 242, 88, 71, 3067, 45, 8
3, 46, 1253, 173, 72, 487, 115, 149, 389, 433, 97, 2235, 2619, 409, 2
94, 426, 435, 39, 233, 1326, 152, 403, 426, 123, 682, 789, 122, 557,
1264, 163, 422, 304, 221, 8342, 1662, 530, 115, 374, 281, 468, 163, 5
56, 30, 70, 1645, 404, 262, 93, 281, 41, 64, 379, 41, 647, 320, 61, 2
53, 73, 192, 361, 713, 120, 483, 2641, 706, 3957, 361, 329, 1476, 42
1, 1029, 761, 521, 333, 94, 399, 167, 976]
120
```

抓取留言數與分享數

使用BeautifulSoup套件，把剛剛找到的 span 標籤及class name 放進 find() 裡面，抓取第一篇貼文的留言數

In [11]:

```
# 抓取所有貼文
soup = Soup(driver.page_source, "lxml")
frames = soup.find_all(class_='gtad4xkn')
```

使用for迴圈，把所有貼文的留言數與分享數分別儲存在各自的list裡

In [12]:

```
# 建立一個空的list
comment_nums = []

# 抓取每一篇貼文的留言數
# class name可能會修改，需要定期偵錯!
for ii in frames:
    read = ii.find('span',class_="d2edcug0 hpfvmrgz qv66sw1b c1et5uql b0tq1wua
    #print(read)
    # 有些貼文沒有留言數，所以抓下來的東西是None，因此直接append 0
    if(read == None):
        comment_nums.append('NULL')
    else:
        comment_nums.append(read.text)
```

In [14]:

```
new_comment_nums = []
for i in range(len(comment_nums)-1):
    if comment_nums[i][-3:] == '則留言' and comment_nums[i+1][-3:] == '則留言':
        new_comment_nums.append(comment_nums[i])
        new_comment_nums.append('0次分享')
    elif comment_nums[i][-3:] == '次分享' and comment_nums[i+1][-3:] == '次分享':
        new_comment_nums.append(comment_nums[i])
        new_comment_nums.append('0則留言')
    else:
        new_comment_nums.append(comment_nums[i])
new_comment_nums.append(comment_nums[-1])
```

In [15]:

```
print(new_comment_nums)
print(len(new_comment_nums))
```

```
['717則留言', '993次分享', '199則留言', '11次分享', '19則留言', '9次分享',
'11則留言', '4次分享', '184則留言', '14次分享', '2則留言', '2次分享', '68則
留言', '6次分享', '25則留言', '8次分享', '94則留言', '13次分享', '30則留言',
'29次分享', '209則留言', '56次分享', '204則留言', '49次分享', '368則留言',
'13次分享', '570則留言', '25次分享', '32則留言', '2次分享', '70則留言', '10
次分享', '19則留言', '23次分享', '261則留言', '59次分享', '150則留言', '53次
分享', '14則留言', '2次分享', '66則留言', '10次分享', '3則留言', '18次分享',
'213則留言', '38次分享', '114則留言', '23次分享', '6則留言', '20次分享', '1
95則留言', '753次分享', '22則留言', '40次分享', '615則留言', '116次分享',
'56則留言', '20次分享', '0則留言', '5次分享', '24則留言', '26次分享', '21則
留言', '5次分享', '62則留言', '3次分享', '518則留言', '146次分享', '642則留
言', '219次分享', '119則留言', '25次分享', '814則留言', '326次分享', '96則
留言', '10次分享', '7則留言', '6次分享', '11則留言', '5次分享', '125則留言',
'158次分享', '15則留言', '1 次分享', '48則留言', '9次分享', '15則留言', '3次
分享', '674則留言', '79次分享', '55則留言', '8次分享', '29則留言', '9次分
享', '302則留言', '39次分享', '2則留言', '8次分享', '95則留言', '10次分享',
'54則留言', '43次分享', '44則留言', '35次分享', '71則留言', '5次分享', '454
則留言', '185次分享', '814則留言', '87次分享', '63則留言', '46次分享', '33則
留言', '81次分享', '99則留言', '40次分享', '330則留言', '38次分享', '0則留
言', '3次分享', '76則留言', '32次分享', '232則留言', '92次分享', '78則留言',
'20次分享', '30則留言', '88次分享', '92則留言', '23次分享', '45則留言', '8次
分享', '11則留言', '93次分享', '45則留言', '33次分享', '20則留言', '13次分
享', '167則留言', '98次分享', '71則留言', '151次分享', '32則留言', '19次分
享', '370則留言', '17次分享', '26則留言', '53次分享', '6則留言', '15次分享',
'600則留言', '2,196次分享', '193則留言', '118次分享', '145則留言', '25次分
享', '2則留言', '5次分享', '91則留言', '78次分享', '198則留言', '15次分享',
'7則留言', '167次分享', '101則留言', '13次分享', '316則留言', '62次分享',
'5則留言', '0次分享', '1 則留言', '9次分享', '222則留言', '150次分享', '179
則留言', '42次分享', '114則留言', '32次分享', '21則留言', '3次分享', '144則
留言', '18次分享', '1 則留言', '3次分享', '20則留言', '9次分享', '155則留
言', '18次分享', '3則留言', '6次分享', '101則留言', '24次分享', '7則留言',
'17次分享', '18則留言', '4次分享', '14則留言', '21次分享', '31則留言', '3次
分享', '16則留言', '17次分享', '125則留言', '56次分享', '169則留言', '23次分
享', '26則留言', '13次分享', '148則留言', '17次分享', '1,440則留言', '778次
分享', '25則留言', '79次分享', '1,653則留言', '454次分享', '70則留言', '50
次分享', '8則留言', '21次分享', '61則留言', '182次分享', '183則留言', '16次
分享', '406則留言', '98次分享', '235則留言', '54次分享', '31則留言', '44次分
享', '382則留言', '30次分享', '38則留言', '21次分享', '26則留言', '54次分
享', '16則留言', '3次分享', '460則留言', '56次分享']
240
```

資料整理

In [16]:

```
# 把'則留言'和','去掉
comments = []
for i in range(len(new_comment_nums)):
    if '則留言' in new_comment_nums[i]:
        index = new_comment_nums[i].find('則')
        comments.append(int(new_comment_nums[i][:index].replace(',','')))
```

In [17]:

```
print(comments)
print(len(comments))
```

```
[717, 199, 19, 11, 184, 2, 68, 25, 94, 30, 209, 204, 368, 570, 32, 7
0, 19, 261, 150, 14, 66, 3, 213, 114, 6, 195, 22, 615, 56, 0, 24, 21,
62, 518, 642, 119, 814, 96, 7, 11, 125, 15, 48, 15, 674, 55, 29, 302,
2, 95, 54, 44, 71, 454, 814, 63, 33, 99, 330, 0, 76, 232, 78, 30, 92,
45, 11, 45, 20, 167, 71, 32, 370, 26, 6, 600, 193, 145, 2, 91, 198,
7, 101, 316, 5, 1, 222, 179, 114, 21, 144, 1, 20, 155, 3, 101, 7, 18,
14, 31, 16, 125, 169, 26, 148, 1440, 25, 1653, 70, 8, 61, 183, 406, 2
35, 31, 382, 38, 26, 16, 460]
120
```

In [18]:

```
# 把'次分享'和','去掉
shares = []
for i in range(len(new_comment_nums)):
    if '次分享' in new_comment_nums[i]:
        index = new_comment_nums[i].find('次')
        shares.append(int(new_comment_nums[i][:index].replace(',','')))
```

In [19]:

```
print(shares)
print(len(shares))
```

```
[993, 11, 9, 4, 14, 2, 6, 8, 13, 29, 56, 49, 13, 25, 2, 10, 23, 59, 5
3, 2, 10, 18, 38, 23, 20, 753, 40, 116, 20, 5, 26, 5, 3, 146, 219, 2
5, 326, 10, 6, 5, 158, 1, 9, 3, 79, 8, 9, 39, 8, 10, 43, 35, 5, 185,
87, 46, 81, 40, 38, 3, 32, 92, 20, 88, 23, 8, 93, 33, 13, 98, 151, 1
9, 17, 53, 15, 2196, 118, 25, 5, 78, 15, 167, 13, 62, 0, 9, 150, 42,
32, 3, 18, 3, 9, 18, 6, 24, 17, 4, 21, 3, 17, 56, 23, 13, 17, 778, 7
9, 454, 50, 21, 182, 16, 98, 54, 44, 30, 21, 54, 3, 56]
120
```

文章內容抓取

In [21]:

```
soup = Soup(driver.page_source,'html.parser')
frames = soup.find_all(class_='ecm0bbzt hv4rvrfc ihqw7lf3 dati1w0a')
```

In [22]:

```
frames[0].text
```

Out[22]:

```
'2021年2月12日，香港電台表示根據中國國家廣播電視總局不允許BBC世界新聞台繼續在中國
境內落地的決定，香港電台將不會再轉播《BBC 時事一周》。歡迎聽眾繼續透過以下渠道支持
本節目：1）訂閱 BBC News 中文的YouTube頻道，節目將於YouTube同步直播：http://b
it.ly/1wkHh5T…… 查看更多'
```

In [23]:

```
contents = []
for f in frames:
    contents.append(f.text)
```

In [25]:

```
print(len(contents))
print(contents)
```

```
120
['2021年2月12日，香港電台表示根據中國國家廣播電視總局不允許BBC世界新聞台繼續在中國境內落地的決定，香港電台將不會再轉播《BBC 時事一周》。歡迎聽眾繼續透過以下渠道支持本節目：1）訂閱 BBC News 中文的YouTube頻道，節目將於YouTube同步直播：http://bit.ly/1wkHh5T…… 查看更多', '歐盟稱，外國記者「工作受到持續騷擾和阻撓，正被趕出中國」，並敦促中國遵守其國際法律義務，確保言論和新聞自由。', '【一周熱點回顧】英國央行公布了最新版50英鎊紙幣設計，上面的新面孔是數學奇才艾倫·圖靈（Alan Turing）。這版紙幣將在今年6月23日，即圖靈生日當天發行。艾倫·圖靈究竟有怎樣的不平凡人生？', '【一周熱點回顧】來自貧民窟的他，因為熱愛舞蹈，跳出了自己的小天地，環遊歐洲，還為法國總統馬克龍表演過！', '【一周熱點回顧】據香港媒體引述\x08香港無綫電視（TVB）表示，不會轉播4月26日舉行的奧斯卡金像獎頒獎禮。無綫外事務科強調，沒有本年奧斯卡金像獎的播映權，而不商討播映權是「純商業考慮」。今年，由挪威導演漢默（Anders Hammer）執導的《不割席》，以香港2019年「反送中」示威為題材，入圍奧斯卡最佳紀錄短片。中國媒體在報道奧斯卡提名名單時，大多沒有提及這部電影與香港示威的關連，並在完整提名名單中，把影片另稱作《不要分裂》。相關報道：…… 查看更多', ' #BBC時事一周 粵語廣播本周內容（香港時間4月4日）： https://bbc.in/3dxxFNb (https://bbc.in/3dxxFNb) | https://youtu.be/MGq-7ed9k74 (https://youtu.be/MGq-7ed9k74) 世界衛生組織公布新冠病毒溯源報告 台灣火車出軌死傷慘重…… 查看更多', '【一周熱點回顧】「他（習近平）骨子裡沒有一點民主」。美國總統
```

文章時間抓取

In [44]:

```
frames = soup.find_all(class_='oajrlxb2 g5ia77u1 qu0x051f esr5mh6w e9989ue4 r7d

frames[0].text.strip('=')
```

Out[44]:

```
'2月13日'
```

In [46]:

```
postimes = []
for f in frames:
    if f.text != '=':
        postimes.append(f.text.strip('='))
```

In [49]:

```
postimes.remove('英國倫敦')
```

In [50]:

```
print(len(postimes))
print(postimes)
```

```
120
['2月13日', '4 小時', '5 小時', '7 小時', '11 小時', '12 小時', '13 小時',
'22 小時', '昨天下午5:00', '昨天下午3:01', '昨天下午2:02', '昨天下午1:01',
'昨天上午11:00', '昨天上午9:01', '昨天上午3:01', '4月2日下午10:54', '4月2日
下午9:00', '4月2日下午8:00', '4月2日下午7:00', '4月2日下午6:00', '4月2日下
午5:15', '4月2日下午5:01', '4月2日下午4:00', '4月2日下午3:31', '4月2日下午
2:15', '4月2日下午1:37', '4月2日下午12:00', '4月2日上午11:43', '4月2日上午1
1:00', '4月2日上午10:00', '4月2日上午9:01', '4月2日上午8:01', '4月1日下午9:
01', '4月1日下午8:01', '4月1日下午7:01', '4月1日下午6:00', '4月1日下午5:0
0', '4月1日下午4:00', '4月1日下午2:01', '4月1日下午1:01', '4月1日下午12:0
0', '4月1日上午11:01', '4月1日上午10:01', '4月1日上午9:02', '3月31日下午10:
46', '3月31日下午10:04', '3月31日下午9:01', '3月31日下午8:01', '3月31日下午
7:56', '3月31日下午7:32', '3月31日下午7:01', '3月31日下午6:22', '3月31日下
午4:20', '3月31日下午3:00', '3月31日下午2:30', '3月31日下午2:00', '3月31日
下午1:00', '3月31日下午12:00', '3月31日上午11:01', '3月31日上午10:00', '3
月31日上午9:02', '3月30日下午9:05', '3月30日下午8:35', '3月30日下午8:01',
'3月30日下午4:44', '3月30日下午2:01', '3月30日下午1:01', '3月30日下午12:0
0', '3月30日上午11:00', '3月30日上午10:00', '3月30日上午9:00', '3月29日下午
11:01', '3月29日下午8:30', '3月29日下午8:01', '3月29日下午7:01', '3月29日
下午6:01', '3月29日下午4:14', '3月29日下午4:00', '3月29日下午3:00', '3月29
日下午2:01', '3月29日下午2:01', '3月29日下午1:00', '3月29日上午11:00', '3月
29日上午10:00', '3月29日上午9:01', '3月28日下午10:11', '3月28日下午5:28',
'3月28日下午5:01', '3月28日下午3:00', '3月28日上午11:59', '3月28日上午11:0
0', '3月28日上午9:30', '3月28日上午9:01', '3月27日下午10:59', '3月27日下午
9:59', '3月27日下午7:59', '3月27日下午5:01', '3月27日下午4:49', '3月27日下
午3:00', '3月27日下午1:37', '3月27日下午1:01', '3月27日上午11:01', '3月27
日上午9:00', '3月26日下午11:31', '3月26日下午9:35', '3月26日下午8:15', '3月
26日下午8:01', '3月26日下午6:07', '3月26日下午5:01', '3月26日下午4:01', '3
月26日下午3:24', '3月26日下午2:03', '3月26日下午12:17', '3月26日上午9:48',
'3月26日上午12:04', '3月25日下午10:04', '3月25日下午8:01', '3月25日下午7:0
1', '3月25日下午4:51', '3月25日下午4:17']
```

In [51]:

```
bbcnews = {}

bbcnews['contents'] = contents
bbcnews['postimes'] = postimes
bbcnews['likes'] = likes
bbcnews['comments'] = comments
bbcnews['shares'] = shares
```

In [52]:

```
bbcnews_df = pd.DataFrame(bbcnews)
```

In [53]:

```
1 bbcnews_df
```

Out[53]:

	contents	postimes	likes	comments	shares
0	2021年2月12日，香港電台表示根據中國國家廣播電視總局不允許BBC世界新聞台繼續在中國境...	2月13日	4187	717	993
1	歐盟稱，外國記者「工作受到持續騷擾和阻撓，正被趕出中國」，並敦促中國遵守其國際法律義務，確保...	4 小時	245	199	11
2	【一周熱點回顧】英國央行公布了最新版50英鎊紙幣設計，上面的新面孔是數學奇才艾倫・圖靈（Al...	5 小時	175	19	9
3	【一周熱點回顧】來自貧民窟的他，因為熱愛舞蹈，跳出了自己的小天地，環遊歐洲，還為法國總統馬克...	7 小時	72	11	4
4	【一周熱點回顧】據香港媒體引述✓香港無綫電視（TVB）表示，不會轉播4月26日舉行的奧斯卡金...	11 小時	354	184	14

In [54]:

```
1 bbcnews_df.to_csv('bbcnews.csv', index=False)
```

In [55]:

```
1 with open('bbcnews.json', 'w', encoding='utf-8') as f:
2     for i in range(len(bbcnews_df)):
3         line = "{'contents': '" + bbcnews_df.contents[i] + "', 'postimes': '" +
4         f.write(line)
```

In [56]:

```
1 driver.quit()
```