數據資料爬取與分析

- requests 模組
  - 發送 GET 請求
    - Response 物件可使用屬性取得不同的回應內容：
      - text：
        - 取得網頁原始碼資料。
        - requests預設的讀取編碼為ISO-8859-1(又稱為 Latin-1)，若讀取的頁面編碼不同，常造成亂碼產生。
        - 可設定 Response 物件的 encoding 屬性，常見設定為：UTF-5、Big5。

- requests 模組
  - 發送 GET 請求
    - Response 物件可使用屬性取得不同的回應內容：
      - content：
        - 取得網站二進位檔案資料。
      - status_code：
        - 取得 HTTP 狀態碼。

數據資料爬取與分析

- requests 模組
  - 發送 GET 請求
    - 讀取網頁原始碼：
      - 以utf-8 編碼讀取網頁的原始碼。
        - https://www.csf.org.tw/main/index.asp
        - https://www.tqcplus.org.tw/index.aspx

- requests 模組
  - 發送 GET 請求

    - 工作方式：
      - import requests 模組
      - 使用 requests.get() 以 GET 方法對指定網址送出請求。
      - 當伺服器機收到後就會回應。

數據資料爬取與分析

- requests 模組
  - 發送 GET 請求
    - 加上 URL 查詢參數：
      - GET 請求可在指定網址後加上 URL 查詢參數，使互動程式接收後導出不同的回應內容。
      - 參數依每個網站結構不同而有差異，如：
        - http://www.test.com/?x=value1&y=value2
      - URL 參數與網址之間要以『?』連接，參數與值要以『=』連接，多個參數以『&』連接。

- requests 模組
  - 發送 GET 請求
    - 加上 URL 查詢參數：
      - URL 參數要使用『字典』資料結構進行定義。
      - 使用 GET 請求時必須將 URL 參數內容設定為 params 參數，即可完成。

數據資料爬取與分析

- requests 模組
  - 發送 GET 請求
    - 自訂HTTP Headers：
      - 在網頁請求中，HTTP Headers 是 HTTP 請求和回應的核心，其中標示了關於用戶端瀏覽器、請求頁面、伺服器...等相關資訊。
      - 在進階的網路資料擷取程式中，自訂HTTP Headers 可以用程式模擬瀏覽器的操作，避過網頁的檢查，是一個常用的技術。
      - 設定方式是在 HTTP Headers 中加入 user-agent 的項目。

- requests 模組
  - 自訂HTTP Headers：
    - 如某些網站頁面，當進行 HTTP 要求時會先檢查操作者是否為瀏覽器，如果不是則無法正常讀取內容。
    - 此時即可利用自訂 HTTP Headers 的方式偽裝為瀏覽器操作，跳過檢查進入網站。
    - 若回應 Response 200，則表示正確成功讀取。
    - HTTP狀態碼：
      - https://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81

數據資料爬取與分析

- requests 模組
  - 發送 POST 請求
    - POST 請求是常見的 HTTP 請求，網頁中有讓使用者填入資料的表單，大多需要用 POST 請求進行傳送。
    - POST 請求常需要加入查詢參數，參數依每個網站結構不同而有差異。
    - POST 傳遞的參數要定義成『字典』資料型態，接著用 POST 請求時必須將傳遞的參數內容設定為 data 參數，即可完成。

- requests 模組
  - 發送 POST 請求
    - 如：自訂payload (字典資料型態) 做為 data 參數向網站提出請求。
      - payload = {'key1': 'value1', 'key2': 'value2'}

數據資料爬取與分析

- requests 模組
  - session 與 cookie 的使用
    - 當用戶端瀏覽器訪問伺服器端時，伺服器會發給用戶端一個憑證以供識別，這個憑證儲存在用戶端的瀏覽器，就是 cookie，產生在伺服器端的就是 session。
    - 當下次再拜訪時，只要所屬的 cookie 與 session 還沒有過期，伺服器就能辨識。
    - 其中所儲存的資料能包含的內容可以很多，如記住網頁登入者的會員資訊，可以使用 Session() 或 cookie 來達成。

- requests 模組
  - session 與 cookie 的使用
    - 建立 session：
      - 建立 session 的語法：
        - rs = requests.Session()
      - 建立的 session 中包含 cookie 資訊，因此可以利用相同的 cookie 對同一個網站的不同頁面提出請求。
    - 在會員制的網站中，會員功能大多需要先登錄認證後才可使用，如果沒有認證，在流程設計上，瀏覽頁面會被先導向會員登錄頁面進行登入，否則無法使用

數據資料爬取與分析

- requests 模組
  - session 與 cookie 的使用
    - 使用 session 請求：
      - 以批踢踢實業坊八卦討論版為例(https://www.ptt.cc/bbs/Gossiping/index.html)，要取得討論列表。
      - 第一次進入時會重新導向到(https://www.ptt.cc/ask/over18)，目的是要確定瀏覽者是否年滿18歲才能進入。
      - 這是一個對使用者資格進行確認的防護機制，對網路資料擷取者是很大的考驗。

- requests 模組
  - session 與 cookie 的使用
    - 使用 session 請求：
      - 因為在資料擷取時，必須經過認證的動作來取得身份才能進行。
      - 一般這樣的機制都必須搭配 session，所以要建立 session，以 post 方式帶著參數進行登錄後，再使用原來的 cookie 以 get 方式帶著參數進入首頁。

數據資料爬取與分析

- requests 模組
  - session 與 cookie 的使用
    - 使用 cookie 請求：
      - 進階網路資料擷取程式中，若目標頁面需要 cookie 值認證，會因為這個機制干擾導致讀取失敗。
      - 解決方式是在進行請求時加入 cookie 值，即可順利的進入目標頁面。
      - 設定方式：
        - 在 request 請求時加入 cookie 參數， cookie 參數必須為字典格式。

- requests 模組
  - Response 物件：
    - 屬性：
      - status_code：回傳的狀態，如果是 request.codes.ok，則表示網頁內容讀取成功。
      - text：網頁內容。

數據資料爬取與分析

- BeautifulSoup 模組
  - 網頁解析
  - 讀取 HTML 原始碼，自動進行解析並產生一個 BeautifulSoup 物件。
  - 此物件中包含了整個 HTML 文件的結構樹，可以利用此結構樹找出網頁中任何的資料。

- BeautifulSoup 模組
  - 可以快速地由HTML中提取內容(對網頁基本結構要有認識)
  - 安裝 BeautifulSoup 模組：
    - pip install –U beautifulsoup4

數據資料爬取與分析

- BeautifulSoup 模組
  - 網頁結構：
    - 純文字(*.html、*.htm)
    - 標籤式語法：<...>
    - 大多數標籤都有起始標籤與結束標籤：
      - <h1>.....</<h1>
    - 每一個標籤代表不同的網頁功能。

- BeautifulSoup 模組
  - 網頁結構：
    -
```
<!DOCTYPE html>
<html>
 <head>
  <title>This is a title</title>
 </head>
 <body>
  <p>Hello world!</p>
 </body>
</html>
```

數據資料爬取與分析

- BeautifulSoup 模組
  - 使用：
    - 利用 requests 模組取得網頁原始碼，再使用 Python 內建的 html.parser 解析原始碼，建立 BeautifulSoup 物件後再進行解析。
      - from bs4 import BeautifulSoup
        html. Parser 物件 = BeautifulSoup(原始碼, '解析方法')

- BeautifulSoup 模組
  - 使用：
    - 解析後在 HTML 中每個標籤都為 DOM 結構中的節點，接著就可以在其中找尋並取出指定的內容。
    - 解析方法：
      - html.parser(常用)
      - lxml(相容性佳，速度快)
        - 使用時需要安裝：
          - pip install lxml
      - html5lib(解析能力強，速度慢)
        - 使用時需要安裝：
          - pip install html5lib

數據資料爬取與分析

- BeautifulSoup 模組
  - 屬性：
    - 標籤名稱：
      - 傳回指定標籤內容。如 sp.title 代表要傳回 <title> 的標籤內容。
    - text：
      - 傳回去除所有 HTML 標籤後的網頁文字內容。

數據資料爬取與分析

› BeautifulSoup 模組

方法	說明(假設建立 BeautifulSoup 物件 sp)
find()	尋找第一個符合條件的標籤，以字串傳回，找不到則傳回 None。如：sp.find(“a”)。
find_all()	尋找(往下搜尋)所有符合條件的標籤，以串列傳回，找不到則傳回空串列。如：sp.find_all(“a”) 尋找指定標籤中符合條件的內容。如： sp.find_all(標籤名稱, {屬性名稱:屬性內容})
find_parent() find_parents()	尋找(往上搜尋)所有符合條件的標籤，以串列傳回，找不到則傳回空串列。
find_next_sibling() find_next_siblings()	在同一層往後尋找特定的標籤
find_previous_sibling() find_previous_sibling()	在同一層往前尋找特定的標籤

數據資料爬取與分析

› BeautifulSoup 模組

方法	說明(假設建立 BeautifulSoup 物件 sp)
select()	尋找指定 CSS 選擇器，如 id、class 的內容，以串列傳回，如：以 id 讀取 sp.select(“#id”)、以 class 讀取 sp.select(“.classname”)。
get()	取得網頁標籤的屬性內容。
get_text()	取得文字內容。
prettify()	顯示 HTML 文件排版結構。