**網頁瀏覽器自動化工具介紹**

個人比較常用的有：

|  |  |
| --- | --- |
| 名稱 | 說明 |
| Nightmare | 一個高階的瀏覽器自動化（測試）函式庫，以 Electron（一種可以開發桌面應用程式的工具，可以開發類似 Windows Form，例如音樂播放器、小算盤等應用程式）為基礎。它可以簡單模仿使用者行為，例如點擊按鈕、換頁、在特定欄位輸入文字等。我們藉此來取得瀏覽器操作過後的 html 元素。  範例程式：  const Nightmare = require('nightmare')  const nightmare = Nightmare({ show: true })  nightmare  .goto('https://duckduckgo.com')  .type('#search\_form\_input\_homepage', 'github nightmare')  .click('#search\_button\_homepage')  .wait('#r1-0 a.result\_\_a')  .evaluate(() => document.querySelector('#r1-0 a.result\_\_a').href)  .end()  .then(console.log)  .catch(error => {  console.error('Search failed:', error)  })  網址：http://www.nightmarejs.org/  GitHub：https://github.com/segmentio/nightmare |
| Selenium | 為使用者設計的一套瀏覽器自動化的工具集，使用「真正的瀏覽器」來模仿使用者行為，透過 WebDriver 來取得瀏覽器的核心功能，例如執行時引用 chrome、firefox等，進而模仿使用者行為來操作瀏覽器，取得互動後的 html 元素。  網址：https://www.seleniumhq.org/ |

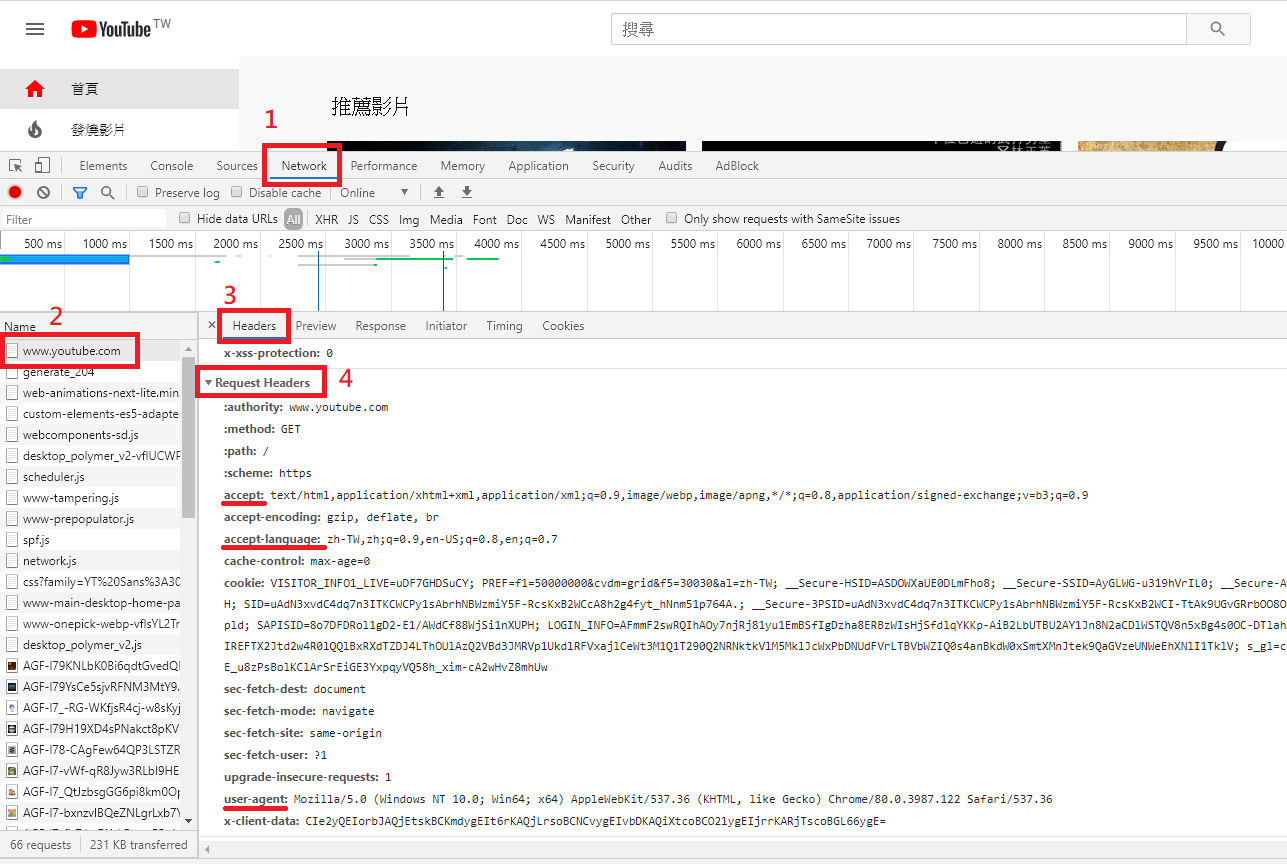
**Nightmare 套件基礎教學**

以下我們會介紹本課程教學中常用的功能：

|  |
| --- |
| 引入方式 |
| //使用 nightmare 套件  const Nightmare = require('nightmare')  //實體化 Nightmare 物件，初始設定為「顯示視窗」、「寬1280」、「高800」  const nightmare = Nightmare({ show: true, width: 1280, height: 800 }); |

|  |
| --- |
| 標頭設定 |
| //瀏覽器標頭，讓對方得知我們是人類，而非機器人 (爬蟲)  const headers = {  'User-Agent':'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3729.169 Safari/537.36',  'Accept':'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3',  'Accept-Language': 'zh-TW,zh;q=0.9,en-US;q=0.8,en;q=0.7',  }; |

標頭設定資訊來源



|  |  |
| --- | --- |
| 方法（物件為 nightmare） | 說明與範例 |
| .goto(網址, headers) | 前往指定網址  await nightmare  .goto('https://www.104.com.tw', headers); |
| .type(選擇器, 輸入文字) | 輸入文字  await nightmare  .type('input#ikeyword', '人帥真好'); |
| .wait(毫秒等待時間) | 讓腳本等待一下，單位為毫秒  await nightmare.wait(2000); |
| .click(選擇器) | 按下哪個 html 元素，例如按鈕、超連結等  await nightmare.click('span#icity'); |
| .catch((err) =>{ 回呼後的程式碼 }) | 補捉到錯誤後的程式區段  await nightmare  .catch(error => {      console.error('Search failed: ', error)  }); |
| .scrollTo( top , left ) | 視窗滾動到指定位置  let offset = 0; //總偏移量  offset += 500;  await nightmare  .scrollTo( offset, 0 )  .wait(500); |
| .evaluate(() => {  執行原生js，並回傳結果  }); | let html = await nightmare.evaluate(() => {    return document.documentElement.innerHTML;  }); |
| .end((err) =>{  回呼後的程式碼  }); | 關閉 nightmare  await nightmare.end((err) => {      if(err) throw err;      console.log('Nightmare is close.');  }); |
| .exists(選擇器) | 確認網頁元素是否存在  await nightmare  .exists('button.b-btn.b-btn--link.js-more-page') |