

D36：加速：非同步爬蟲



簡報閱讀



範例與作業



問題討論

加速：非同步爬蟲



本日知識點目標



加速爬蟲



非同步爬蟲



Python 中實現非同步



加速：非同步爬蟲



解題時間

>

Day 36 提升爬蟲程式執行效率

加速：非同步爬蟲



出題教練：張維元



1

本日知識點目標



本日知識點目標

- 了解非同步爬蟲加速原理與實作

當資料量龐大或是更新速度較為頻繁的狀況下。依照正常的爬蟲程式，可以會因此受到應用上的限制。所以必須用程式的方法，來思考如何加速爬蟲的處理速度。

非同步爬蟲

第二種加速的方法是「非同步爬蟲」，一般程式都需要等前一行執行完畢之後才會執行下一行，而非同步爬蟲的作法則是當某一行程式開始執行時（不用等到結束）就繼續執行下一行。

Python 中實現非同步

```
1 import aiohttp
2 import asyncio
3
4 async def fetch(session, url):
5     async with session.get(url) as response:
6         return await response.text()
7
8 async def main():
9     async with aiohttp.ClientSession() as session:
10         html = await fetch(session, 'http://python.org')
11         print(html)
12
13 loop = asyncio.get_event_loop()
14 loop.run_until_complete(main())
```

加速爬蟲: 異步加載 Asyncio 莫須

#4.2 加速爬虫: 异步加载 Asyncio (爬虫 scraping 基础 Python)



圖解非同步爬蟲與套件的用法

解題時間



解題時間 LET'S CRACK IT

Sample Code & 作業

開始解題



下一步：閱讀範例與完成作業