== 工具 ==

python3 (3.10+), seleniumwire, undetected\_chromedriver

== 網站 ==

www.partslink24.com

== 說明 ==

1. 從入口網站左下方點選 bmw 徽標進入車型點選畫面

2. 從車型頁面源碼中找到相對應的車型 API 接口，進行呼叫。目的是要模擬點選

X1 -> F48 -> SAV -> X1 18d -> THA > Engine

3. 每一個點擊都有相對應的接口，和 json 返回內容。通過這些 json 找到下一層的 API 並獲得 json 內容。

=== 第一题 ===

在 partslink 下，一個汽車品牌下有多個車型年款，由不同的參數組裝而成.

掃描獲得 BMW 下所有車型，並保留每一個接口對應的 json 內容

e.g.

https://www.partslink24.com/p5bmw/extern/groups/func-mdl?hg=04&lang=zh-TW&mdl=X1&modelType=F48&res1=SAV&res2=X1+18d&res3=THA&serviceName=bmw\_parts

=== 第二题 ===

汽車品牌數據是樹狀結構。以BMW為例，

BMW - 1'

- E81

- 3 Doors

- 116d

- ECE

- 11 Engine

- 11\_4021 Short Engine

- Part 1

- Part 2

- 11\_4023 Engine block

- 11\_4099 Engine block mounting parts

- E82

- E87

由於車型眾多，為了加速數據獲取，一般都會以分散式框架，平行獲取。

在第一題 我們知道n+1層數據的API 都可以在 第n層 的API裡獲得。

假設獲得的API URL 會在 30分鐘後過期，且有五隻爬蟲實例，每個爬蟲每次可以發送一個 URL 請求，

請問該如何規劃請求來達到最少重複請求(最少請求次數)

題目補充：

1、完成後，請放 python code 以及結果放在 github 上，如果沒有 github 賬號的話，zip 後交給 HR；

2、不是透過 selenium + browser 點選連結，而是在進入主網頁後 使用 seleniumwire 自帶的 get/post 來進行 API 呼叫。