

利用 POSTMAN 中的 collection 來快速測試 application 功能是否正常

 blog.yowko.com/2017/03/postman-collection-runner.html

無意中聽到 frontend team 的同事在抱怨 frontend 畫面出現錯誤提示，得花點時間查 log 才能發現是 api 有狀況。當下聯想到之前很多專案也都缺乏 api 的監控機制，常見的是 MIS 針對 server 進行 cpu、memory、hd 甚至網路流量進行監控，當然有愈來愈多注重服務的團隊會進行 application 等級的心跳監控，但 application 功能愈來愈多，監控項目是不是也隨著增加呢？

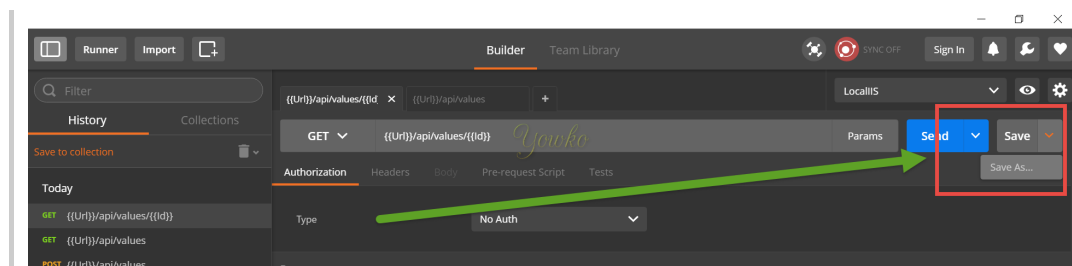
同事討論的問題發生原因也是這麼來的，application 本體正常運作，但因為新功能 api 未回傳正確資料使得 frontend 部份功能異常，所以出現監控系統回應正常但有些功能卻無法 work 的問題

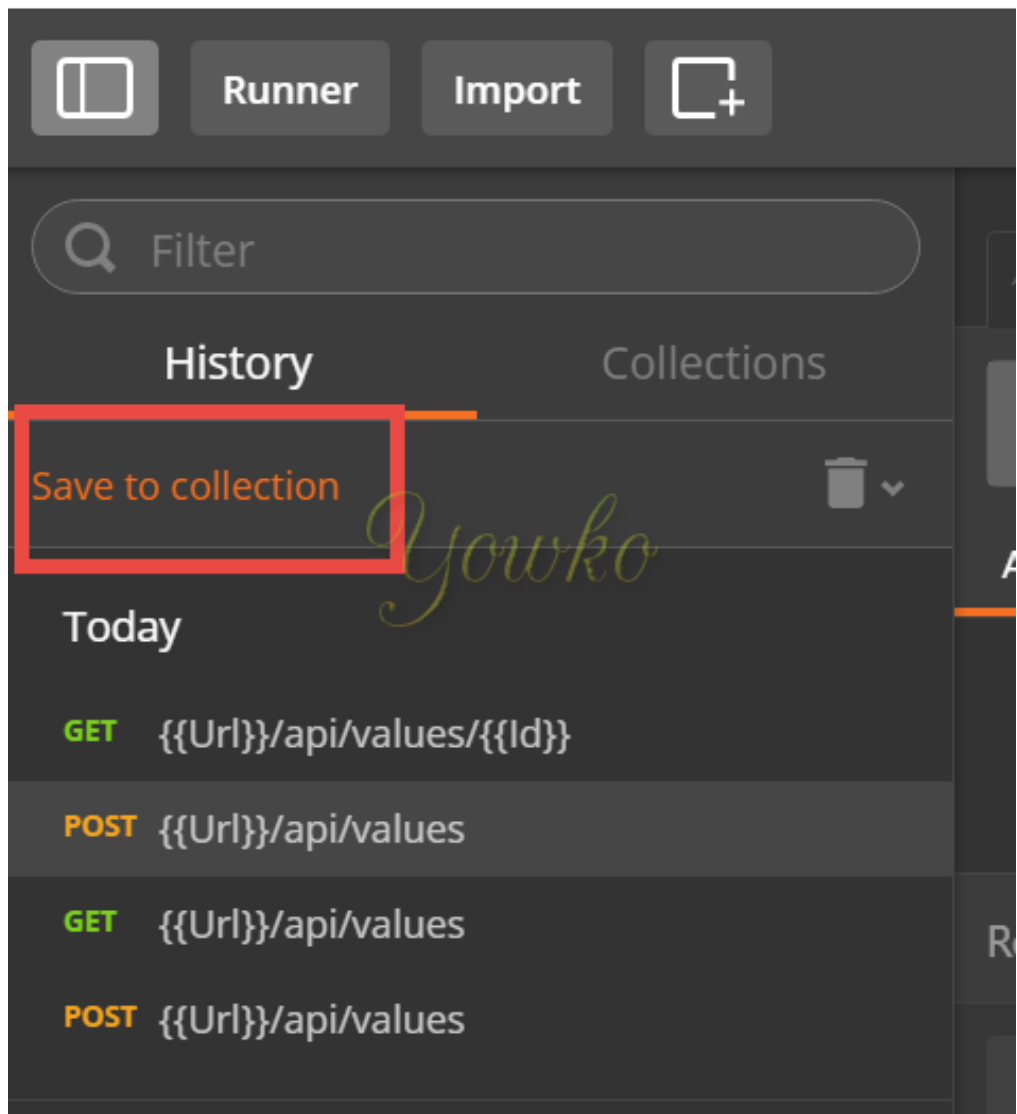
在跨團隊交接時，postman 的 collection 是個方便的溝通方式(當然 swagger 也是)，今天就來介紹如何利用 postman 的 collection 做基本的測試，可以當做快速定位異常的方法之一

準備 collection

將需要基本監控的 request 準備專屬的 collection，當然也可以將開發用的 collection 拿來用，但需要注意不同環境間的參數是否可通用，建議使用參數化寫法，可以參考 [如何在 POSTMAN 中使用參數化寫法來進行不同環境的測試](#)，我就延用其中的範例來進行後續的 demo

1. 逐一將 request 加入 collection 中
 - 按下 **Save** or **Save to collection**





- 填寫基本資料
 - 為 request 命名(使用好辨識的名稱)
 - 加上 request 相關描述(optional)
 - 加入已存在的 collection 或是 建立新的 collection

SAVE REQUEST

Request Name

1 使用 id 來取得資料

Request description (Optional)

2 使用參數化環境設定中的 id 來取得資料

Yowko

Descriptions support Markdown

Save to existing collection / folder

Type to filter

▼

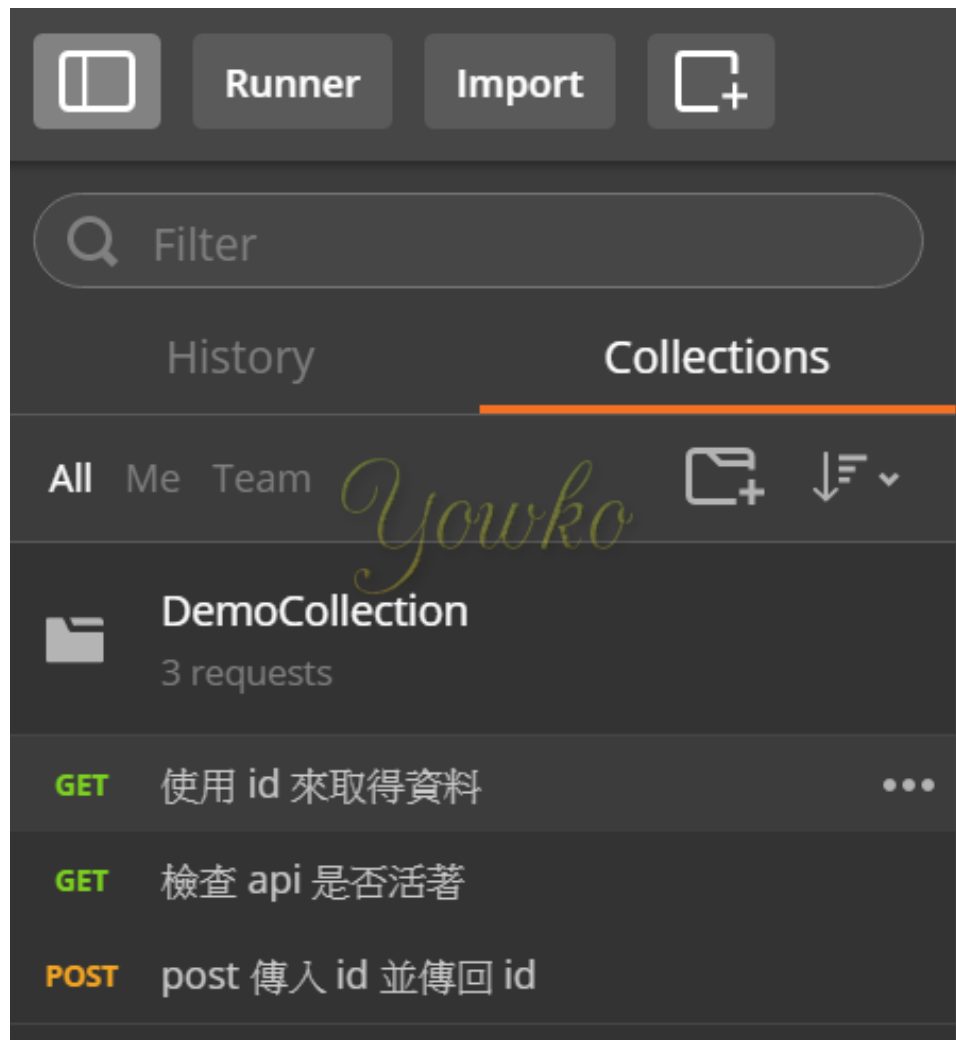
3 Or create new collection

DemoCollection

Cancel

Save

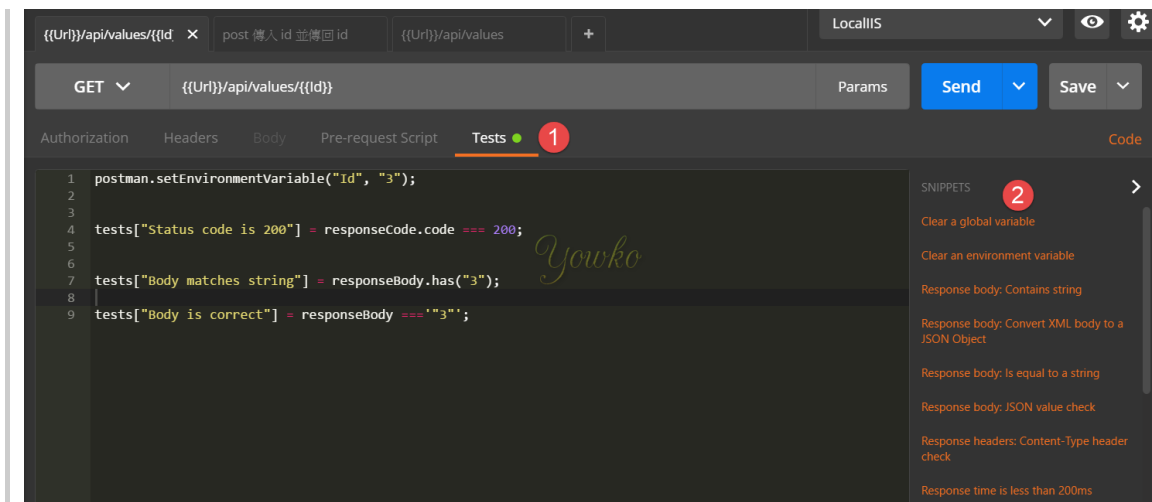
2. 重複上個步驟直到所需 request 皆已加入為止



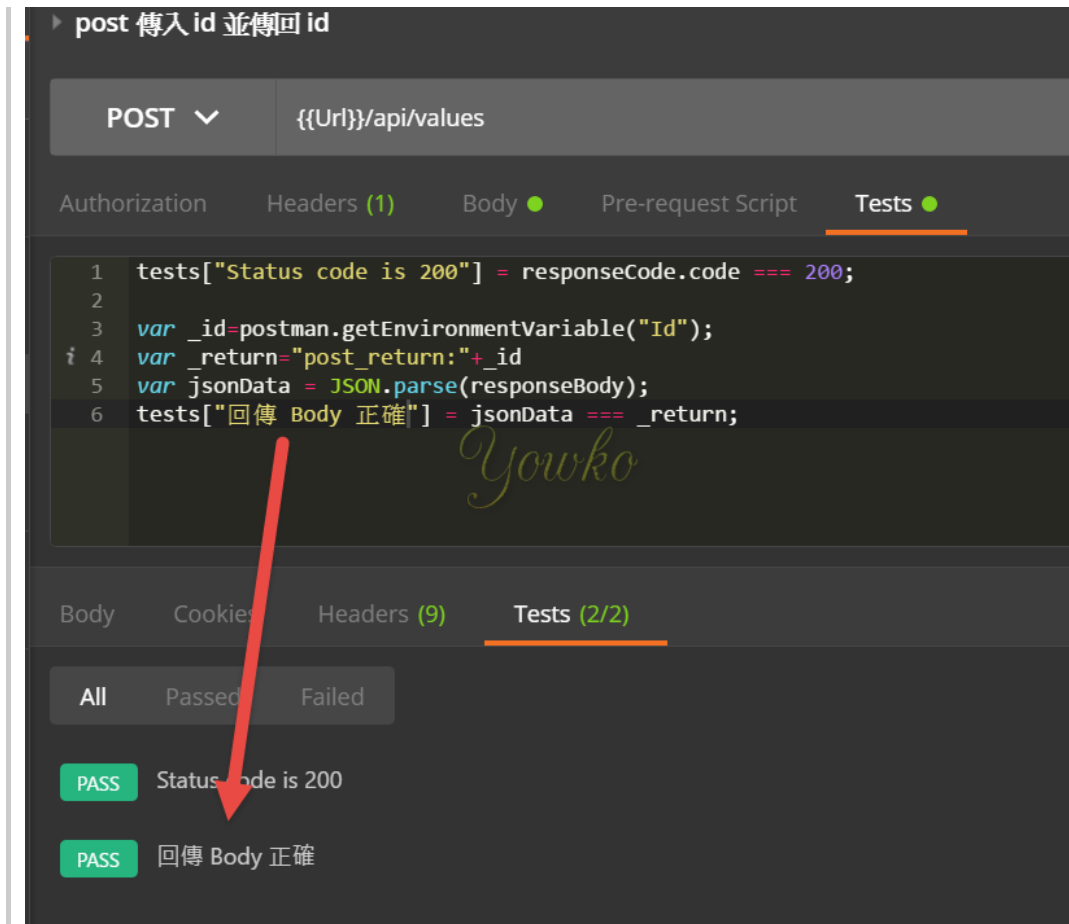
為每個 request 寫驗證邏輯

沒有寫驗證會直接被判定為 pass，這樣就失去執行驗證測試的意義了

1. 開啟 request 的 **Tests** tab
2. 依實際情境撰寫 response 驗證邏輯
 - 有 snippets 可以輔助撰寫
 - 驗證方式可以參考 [Testing examples](#)



3. 測試名稱可以自訂 讓測試結果更一目瞭然



4. 實際範例

- 指定環境變數

```
postman.setEnvironmentVariable("Id", "3");
```

- 驗證 response 的 http status code 為 200

```
tests["Status code is 200"] = responseCode.code === 200;
```

- 驗證 response body 中包含 "3"

```
tests["Body matches string"] = responseBody.has("3");
```

- 驗證 response body 等於 "3"

```
tests["Body is correct"] = responseBody === "3";
```

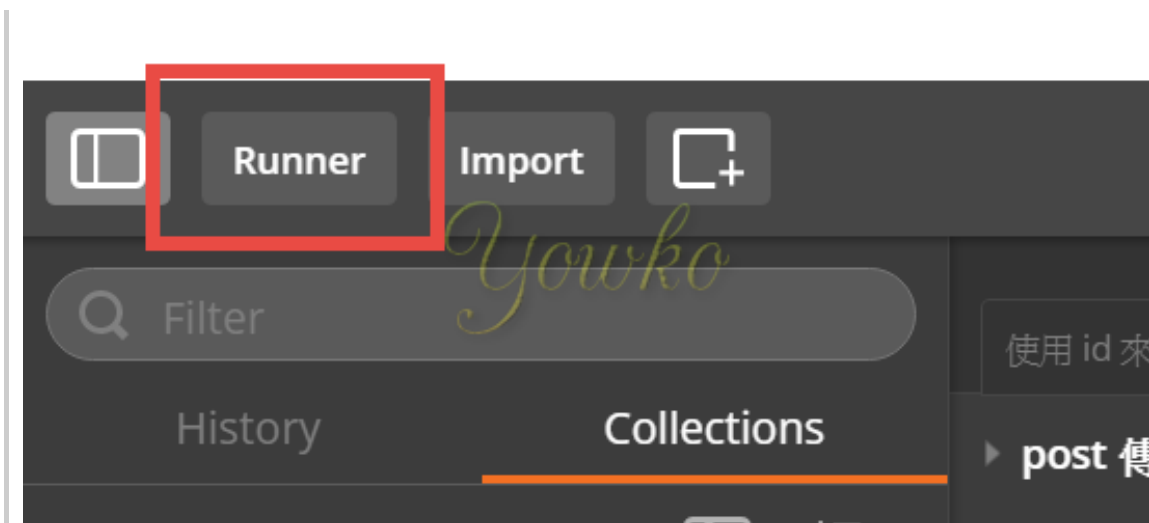
- 驗證 response body 的內容是一個字串

```
var _id=postman.getEnvironmentVariable("Id");
var _return="post_return:"+_id
var jsonData = JSON.parse(responseBody);
tests["Body is correct"] = jsonData === _return;
```

可以透過 `console.log` 的方式來為撰寫的 test 偵錯，設定方式請參考 [如何對 POSTMAN desktop app 偵錯](#)

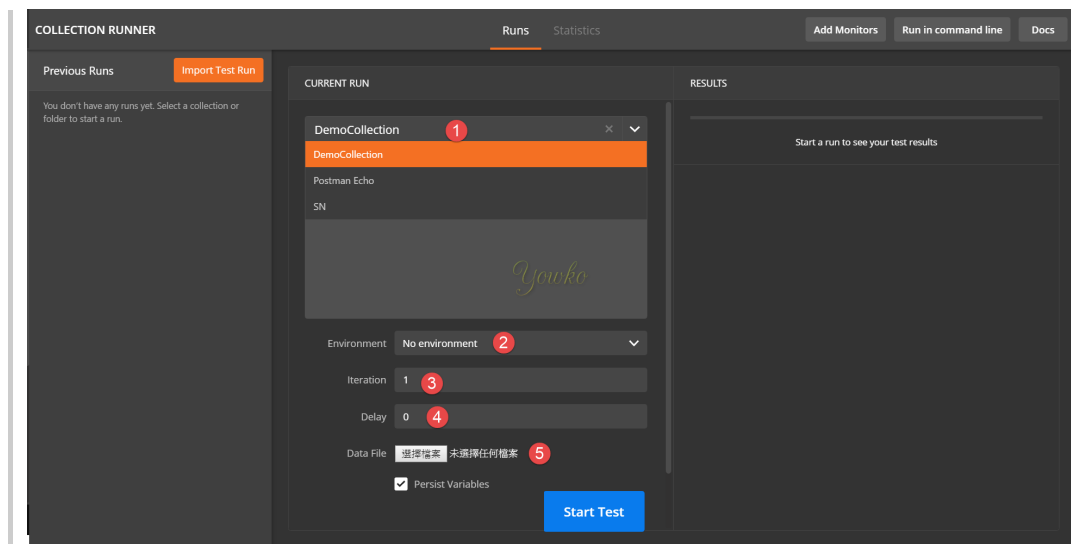
使用 postman runner 來執行測試

1. 開啟 postman Collection Runner



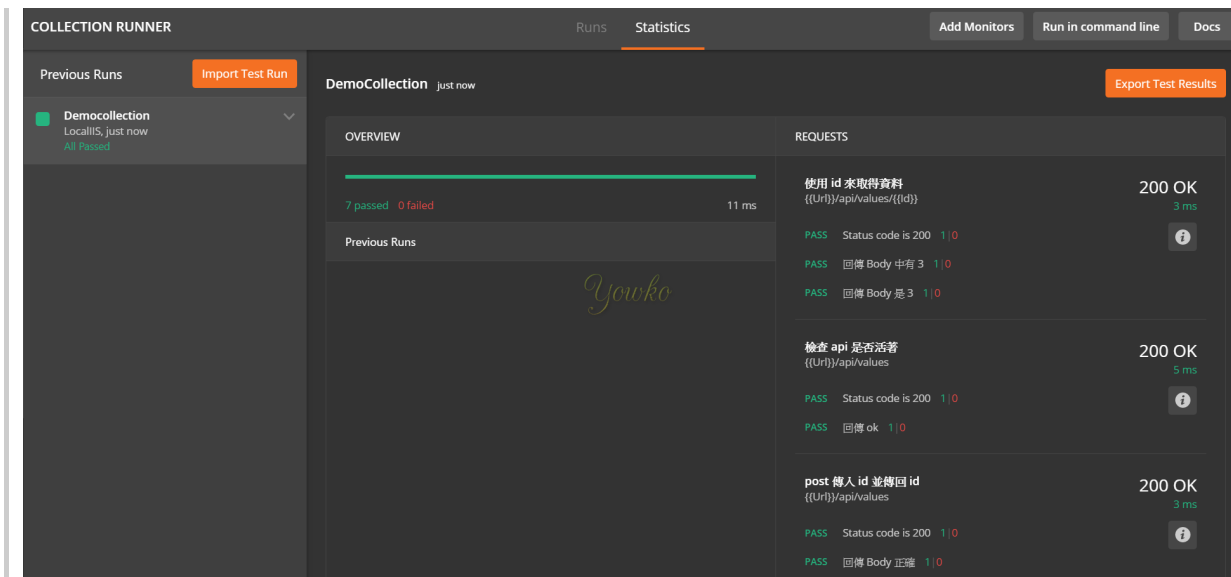
2. 設定 Runner

- 選擇要執行的 collection
- 指定環境變數
- 指定執行次數
- 指定每個 request 間的時間隔毫秒數
- 指定資料來源 可以使用 CSV/JSON



3. 按下 Start Test 開啟測試

測試結果



參考資料

1. [Running a collection](#)
2. [Testing examples](#)
3. [如何對 POSTMAN desktop app 偵錯](#)