Callback function

& IIFE

# 前言

之前介紹過了function基本觀念，不管是宣告方式、閉包或是需要注意的hoisting，今天也就來講講，關於callback的另外一個故事

# Callback function

什麼叫做Callback functon，~~就是把function 叫回來~~就是把函式當作另外一個函式的參數，透過另外一個函式呼叫他，就叫做Callback function。

像是之前我們說過的監聽事件

addEventListener(‘click,function(){},false)就算是callback function的一種

## 把函式當作參數

「函式當作另外一個函式的參數，透過另外一個函式呼叫它」

除了剛剛說的監聽事件外，還有一個經典的案例

Window.setTimeout(function(){},毫秒數)

如果我們希望某段時間之後執行某件事情，就可以透過wondow.setTimeout來協助我們達成，上面那段CODE的轉換成中文的意思就好像是:「在經過毫秒數後，執行function()」

第一個參數是我們想要做的「事情」，而第二個則是設置我們需要「間隔多久」之後才執行，不過要注意時間是用毫秒數計算，不是秒數(1000毫秒 = 1秒)

當然兩個範例的callback function也可以單獨抽出來，會使程式碼看起來能夠更直觀清楚

CODE

除了以上兩種場景會用到Callback function之外，還有一場景會用到 => 「控制多個函式的執行順序」

比如說我們有兩個function

CODE

因為程式碼是一行一行的往下執行，兩個function也是立即執行的，所以結果一定是

Result

當我們加上隨機執行時間之後

CODE

我們就會沒法確定是funcA 還是 funcB先執行了，但如果同時想要「隨機執行時間」與「先執行A在執行B」我們要怎麼做到呢?

這種時候我們就可以透過Callback function來處理

CODE

覺得很難理解的話，我們先不要管function，我們只需要能夠確確實實的console出functionA -> functionB的順序，除了寫成兩個function，我們也可以這麼做(寫成一個function)

CODE

再來我們想要延遲些時間在執行這個function,所以我們可能這麼做

CODE

這個時候我們把console.log(‘funciton B’)來出來，獨立寫成一個function (funcB)，並且傳入到第一個func當作參數來呼叫，(當然也可以將funcB值也寫在第一個func裡面，但為了通用性使用的話，用參數會更好)

CODE

是不是跟我們剛剛寫得一模一樣了，我們再把第一個function改叫做funcA，並且在內部做一個判斷(確認使用的參數是一個function)，爾後也幫funcB加上setTimeout，這樣就完成了

CODE

使用callback functoin時要注意，如果相依性過深，會造成「波動拳」很難維護

CODE

# IIFE

之前function筆記裡面我們有提到過不同宣告的function，其中有一種是匿名function，現在我們就來講講這個fucntion吧

## 當迴圈遇到function

假如我們想要寫一個迴圈，必須透過setTimeout印出1~5，我們可能會這麼做

CODE

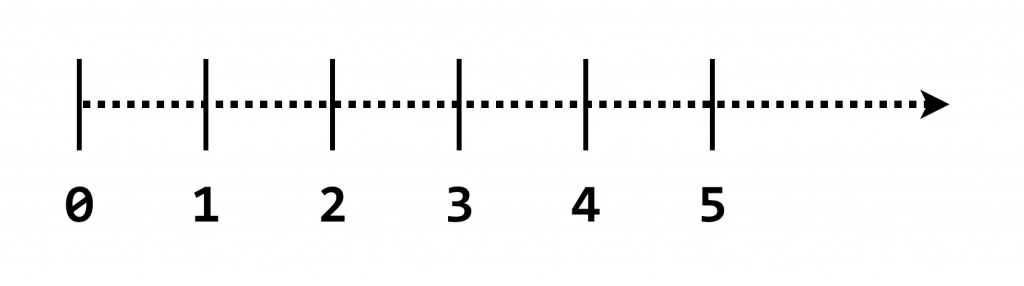
我們就會發現結果直接印出了5個5，哈哈哈哈哈哈哈哈(瘋了)

結果圖片

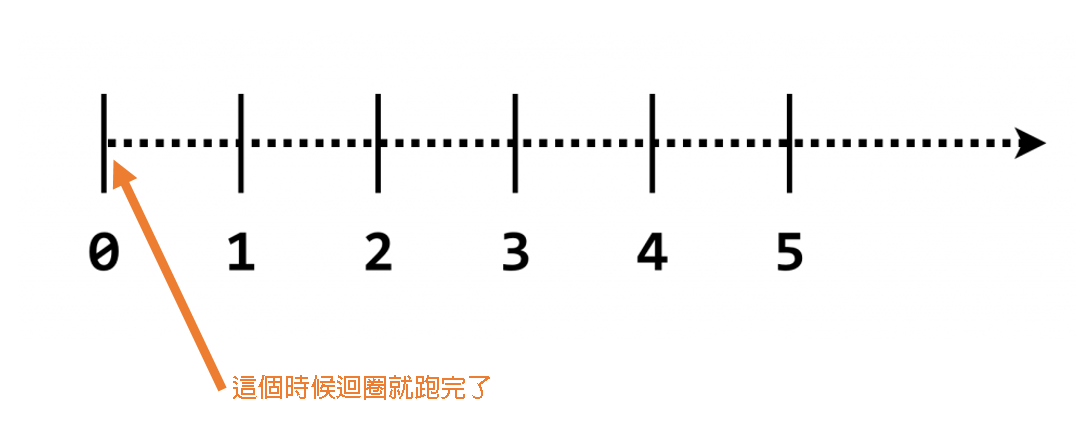
問題出在哪裡呢?

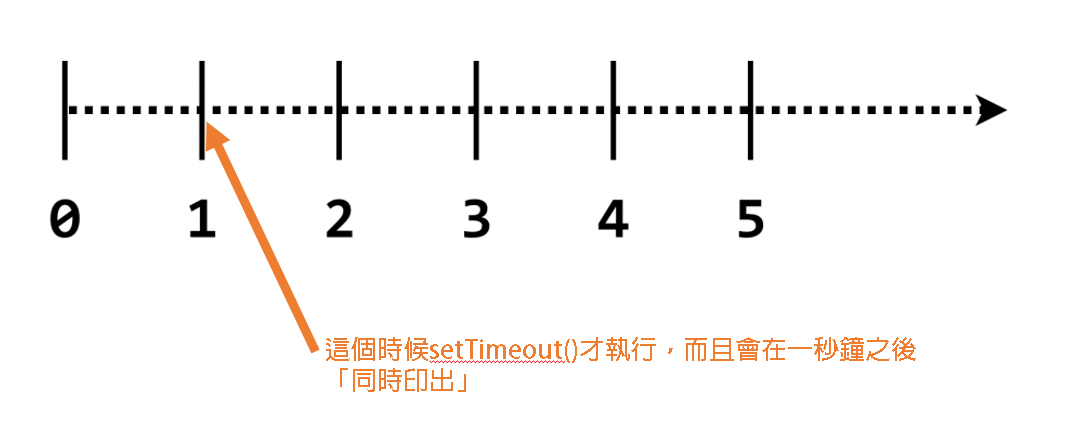
之前介紹變數的時候有提過

切分var變數的範圍是用fucntion來切割的，所以假時間線是這樣



我們知道javascript是一個「非同步」語言，也就是迴圈不會等setTimeout跑完才繼續迴圈，而是一口氣跑完



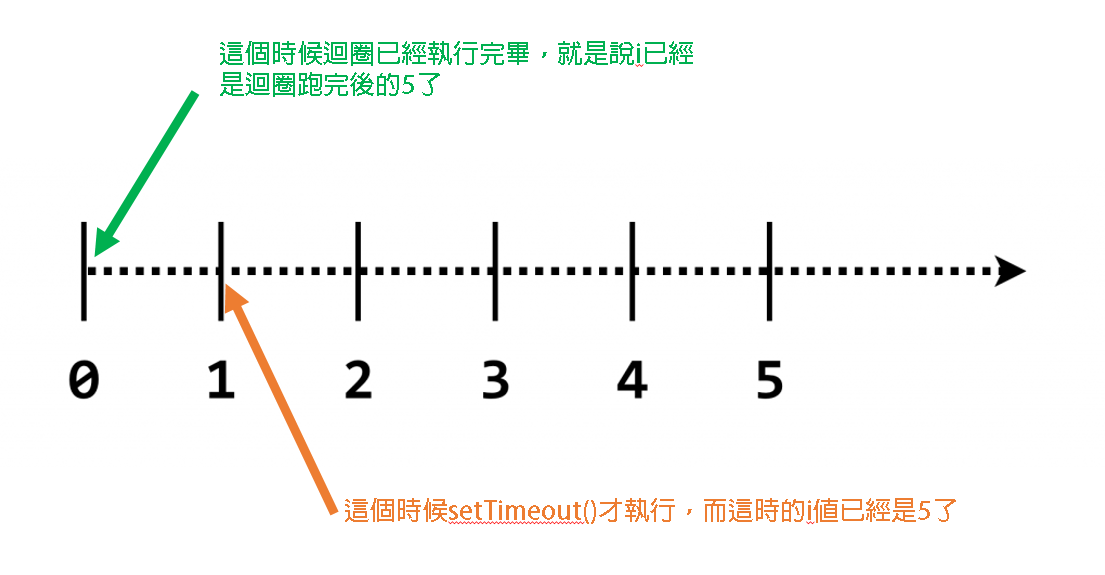


這裡有兩個問題需要去解決

* Console出的i
* Console的執行時間

CODE

這裡的i是從函式「外層」取得的，也就是說迴圈不會等到setTimeout跑完才繼續跑，而是一口氣跑完，而迴圈跑完後setTimeout執行取得的i是已經迴圈的i



所以要怎麼解決這個問題呢?利用var的有效範圍是在function內囉!也就是用我們這裡要介紹的IIFE

(function() {

})();

看不懂嗎?沒關係，我們來一一拆解這個結構

1. 普通的function

function doSomething(){

}

1. 用括號刮起來

(function doSomething(){

})

1. 要呼叫它，所以在最後面再加一個括號，把名字去掉

(剛剛完成的)();

一般的呼叫:funcA(123);

宣告的當下就呼叫:

這就是我們所看到的IIFE拉(Immediately Invoked Function Experssion)「立即被呼叫的特殊涵式」

回到剛剛的迴圈，我們就可以來使用IIFE拉

CODE

這樣子數字就可以按照我們想的[1,2,3,4,5]拉!

不過還是都在一秒鐘後同時出現，這個問題就簡單多了，因為我們setTimeout的時間都是設置一秒阿!所以將其改掉即可

CODE

上面我們用了一個經典的案例來跟各位介紹 IIFE 的用法，除了在迴圈內呼叫 function 我們會需要用 IIFE 來把參數的值保留起來之外 [註1]，IIFE 還有另外幾個好處，就是可以**減少「全域變數」的產生，同時也避免了變數名稱衝突的機會**。

如果你有去看過 jQuery 的原始碼，就會發現 jQuery 也用了相同的手法將 window 與 undefined [註2] 保留起來：

(function( window, undefined ) {

// 略...

})( window )*;*

※ES6之後新增了let 與 const，使用了{}來當作它的scope，也就是剛剛在for中我們使用let的話，就可以做到保留i在執行迴圈當下的值囉!也很建議以後都使用let 與 const，會省掉很多的麻煩!!

[註2] undefined 是可以被修改的 (詳見 [重新認識 JavaScript: Day 03 變數與資料型別](https://ithelp.ithome.com.tw/articles/10190873) ) ，所以 jQuery 雖然在 IIFE 定義了兩個參數，但只傳了一個 winodw，就是為了保持 undefined 原本的樣子。