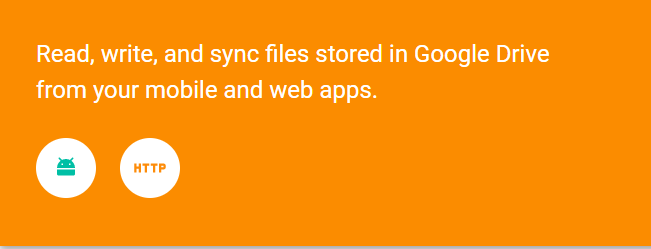
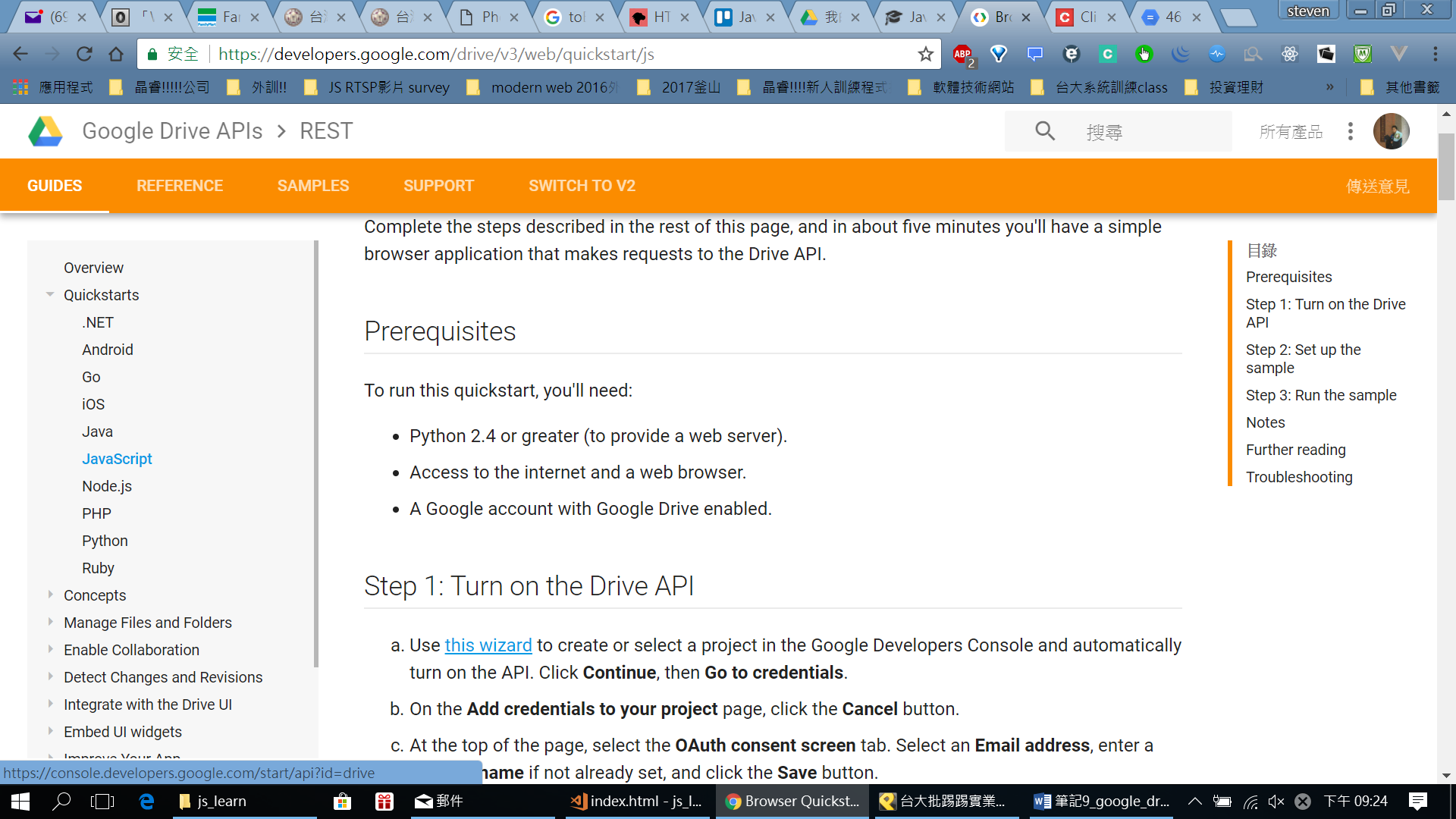
1. Google drive就是模擬像在你本機:
2. Google drive API: <https://developers.google.com/drive/>
3. 有分android, http
4. 選擇Quickstart:

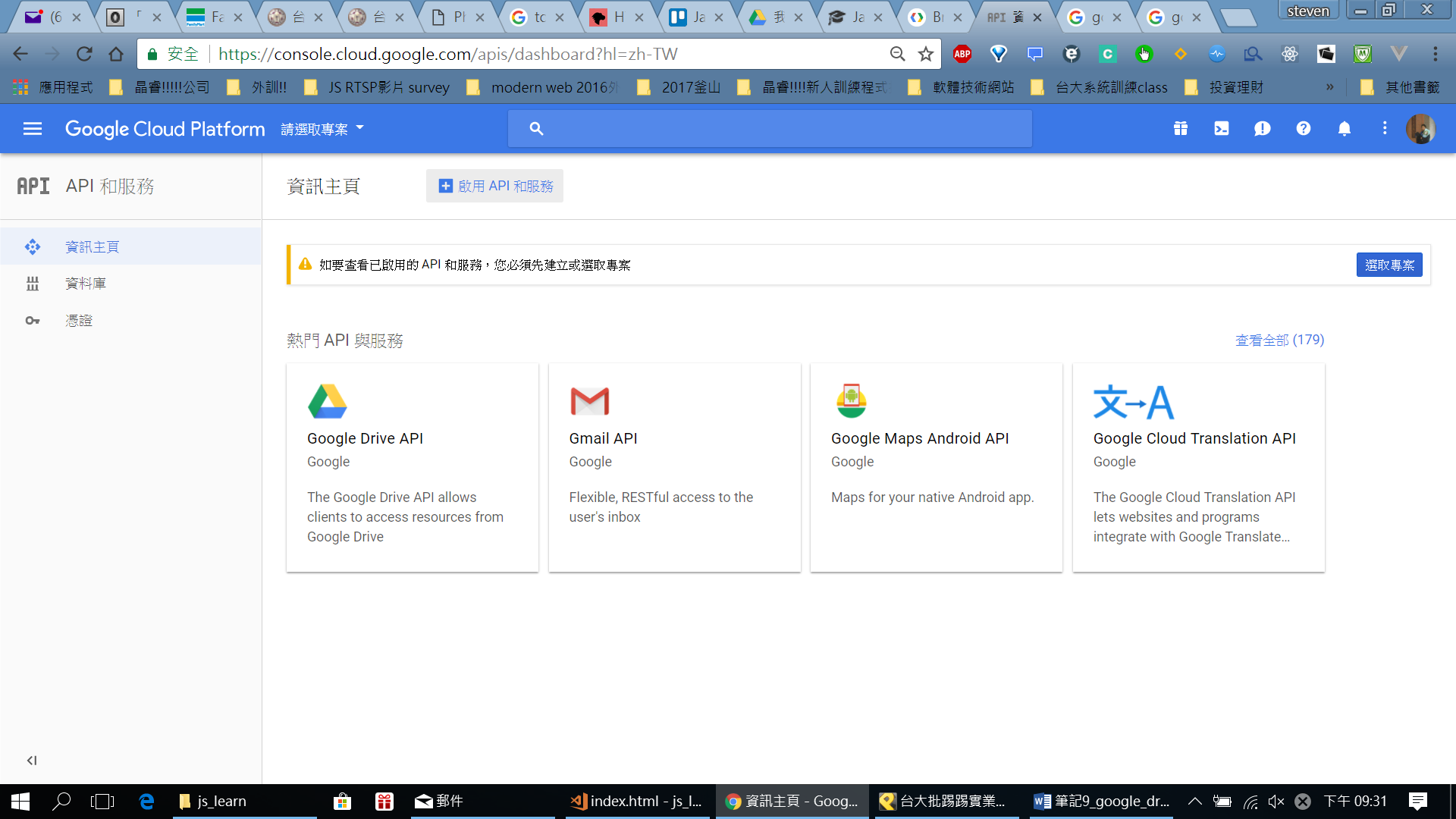
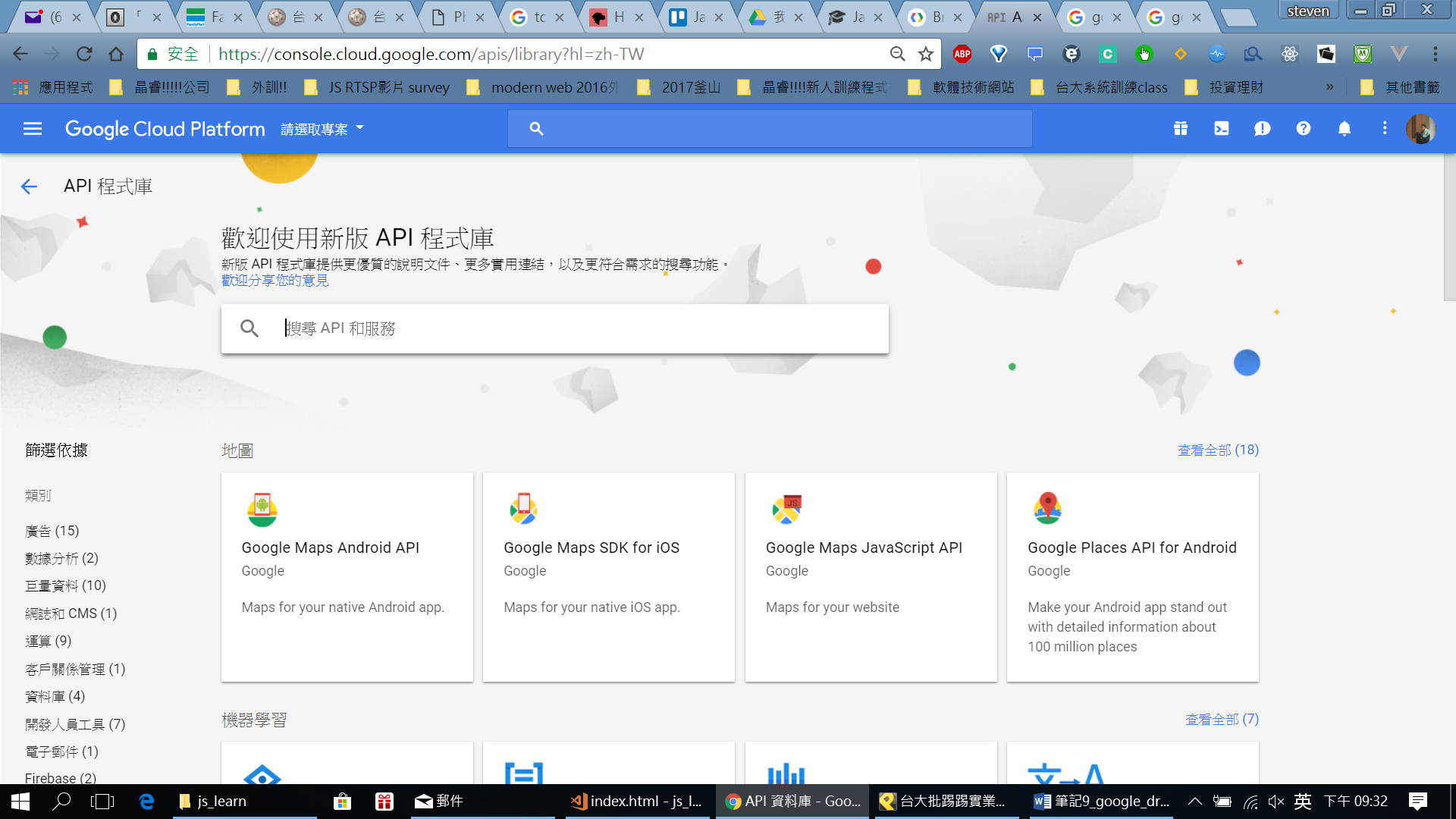
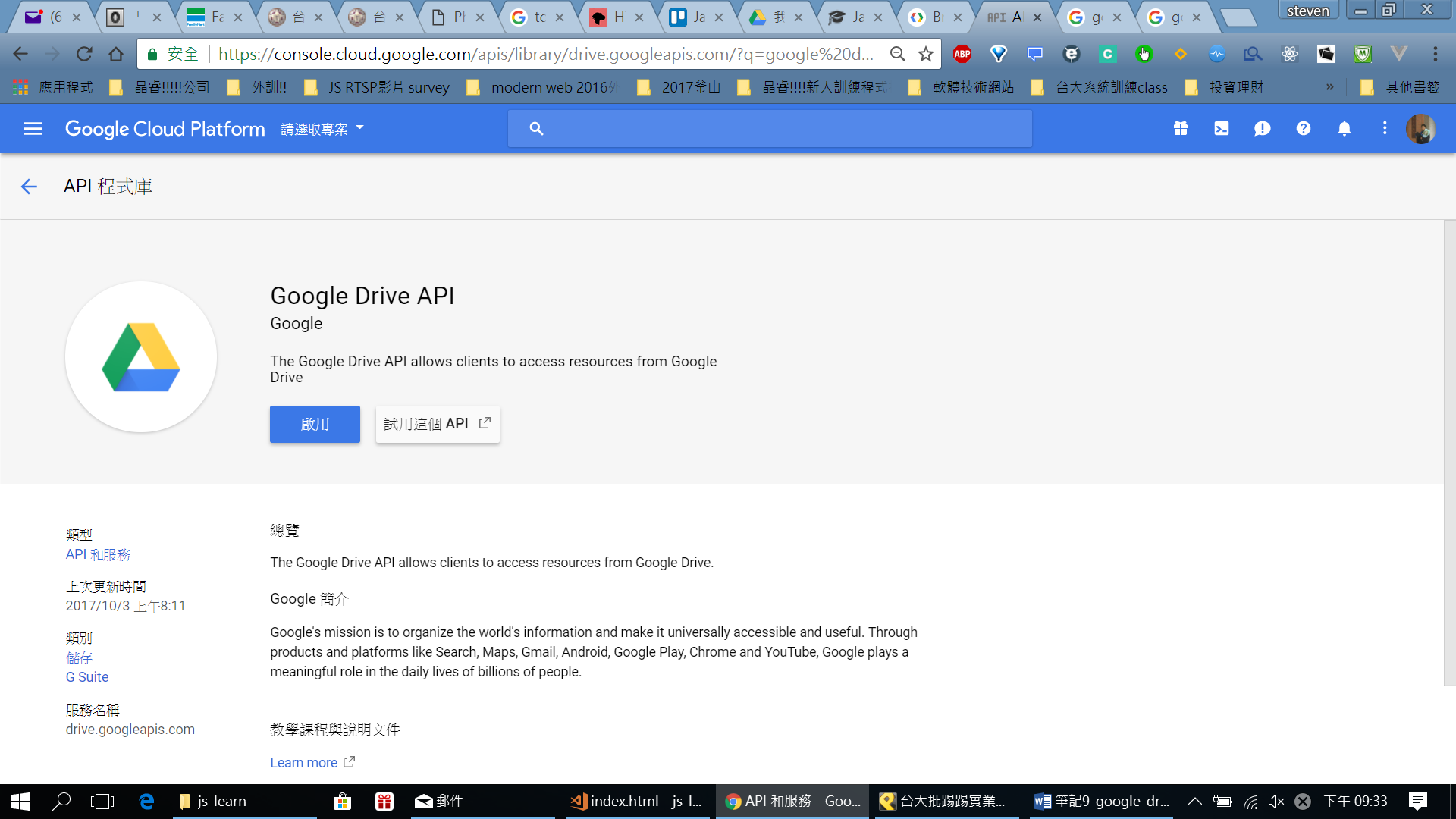
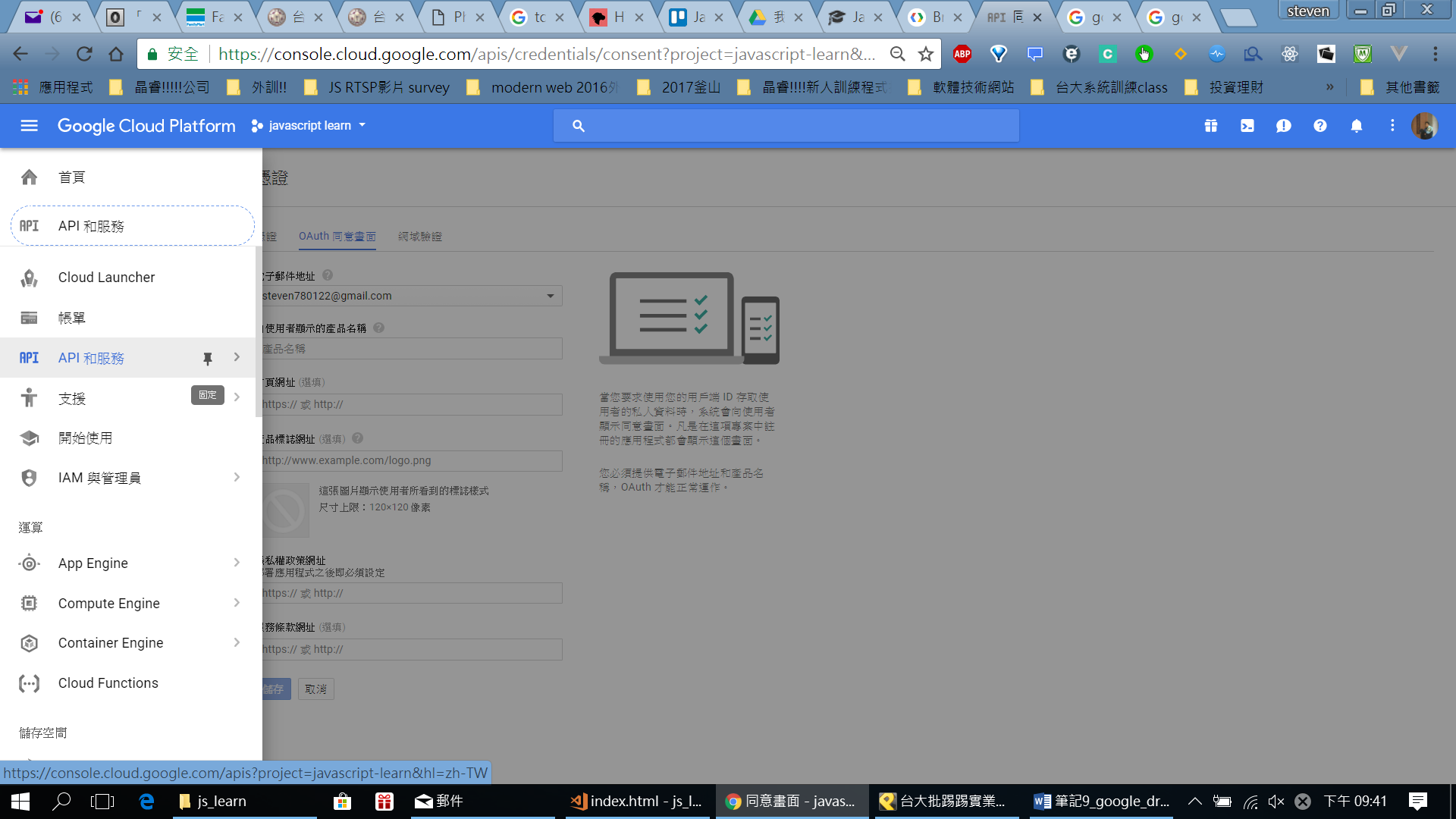
不一定要python 那是他提供Web server 我用nodeJS OK



不要點this wizard

1. Google cloud console: <https://cloud.google.com/cloud-console/>

<https://console.cloud.google.com/cloud-resource-manager?previousPage=%2F%3Fhl%3Dzh-TW&hl=zh-TW>

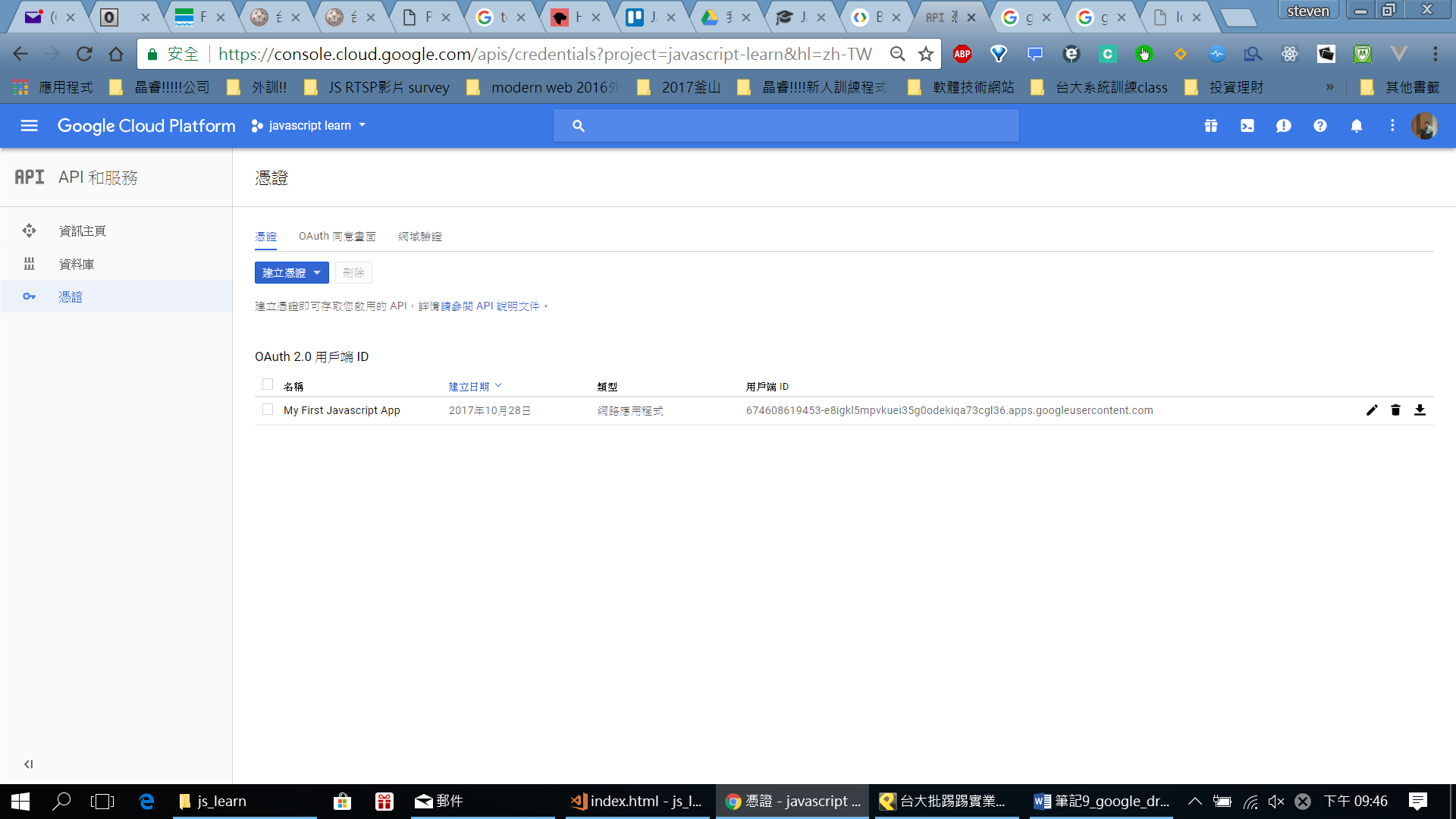
1. 
2. 點選啟用API
3. 
4. 
5. 
6. 把它啟動:



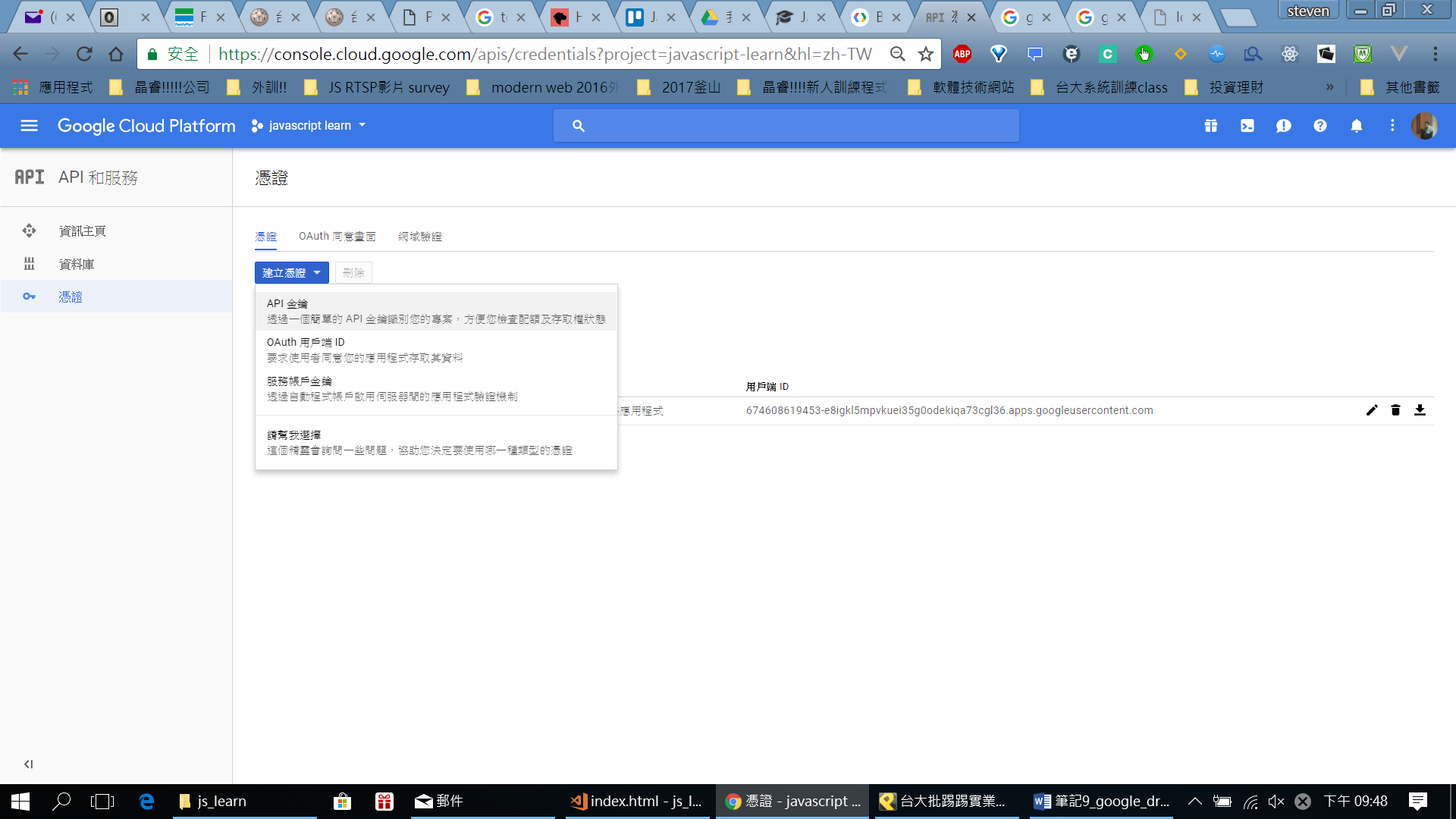
1. 選左邊憑證
2. 選OAuth 憑證(google drive會用到)
3. 要先建立產品名稱的話>>
4. 選網頁應用程式:



輸入localhost:3000 (nodeJS我的server) <http://localhost>

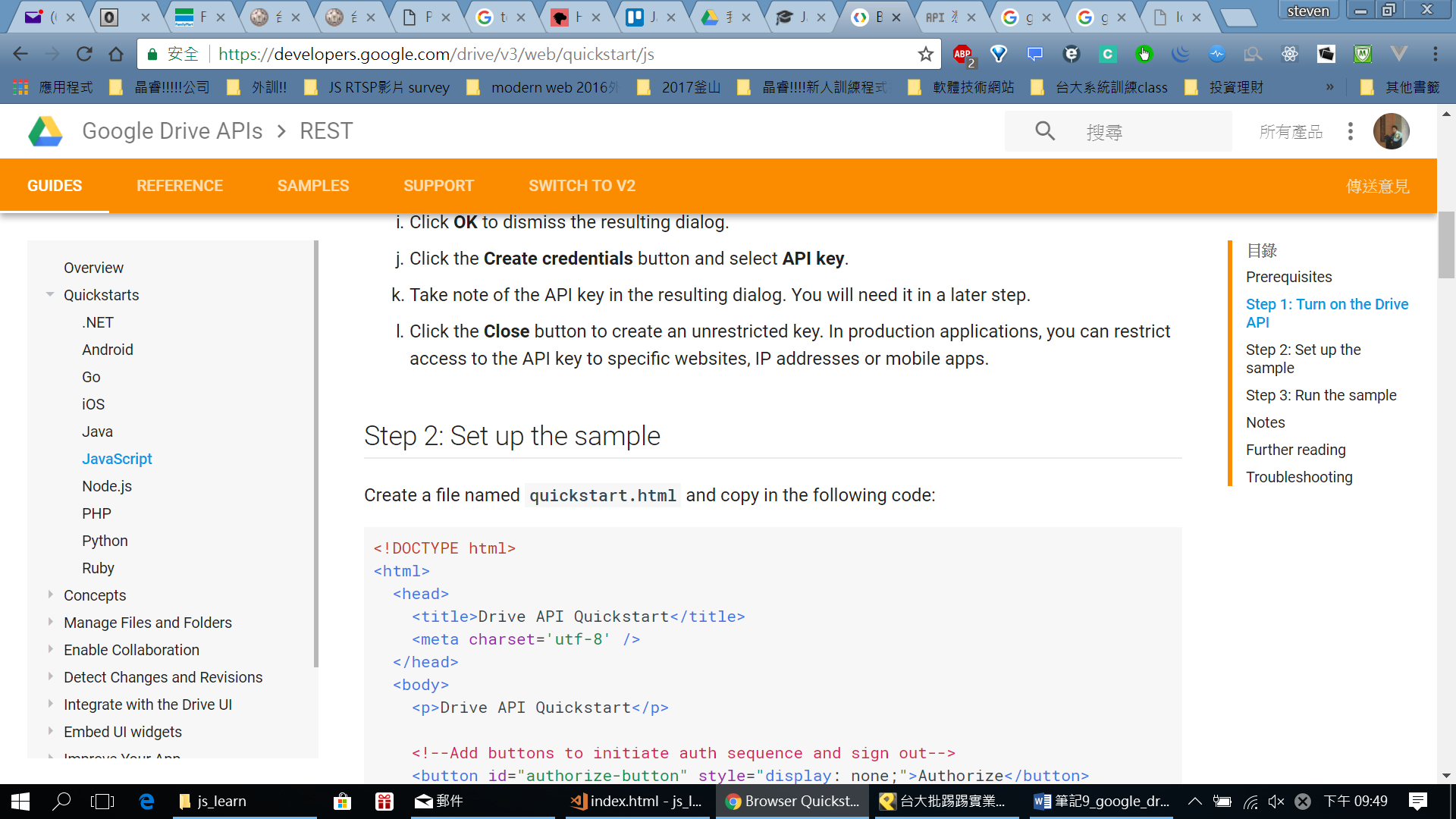


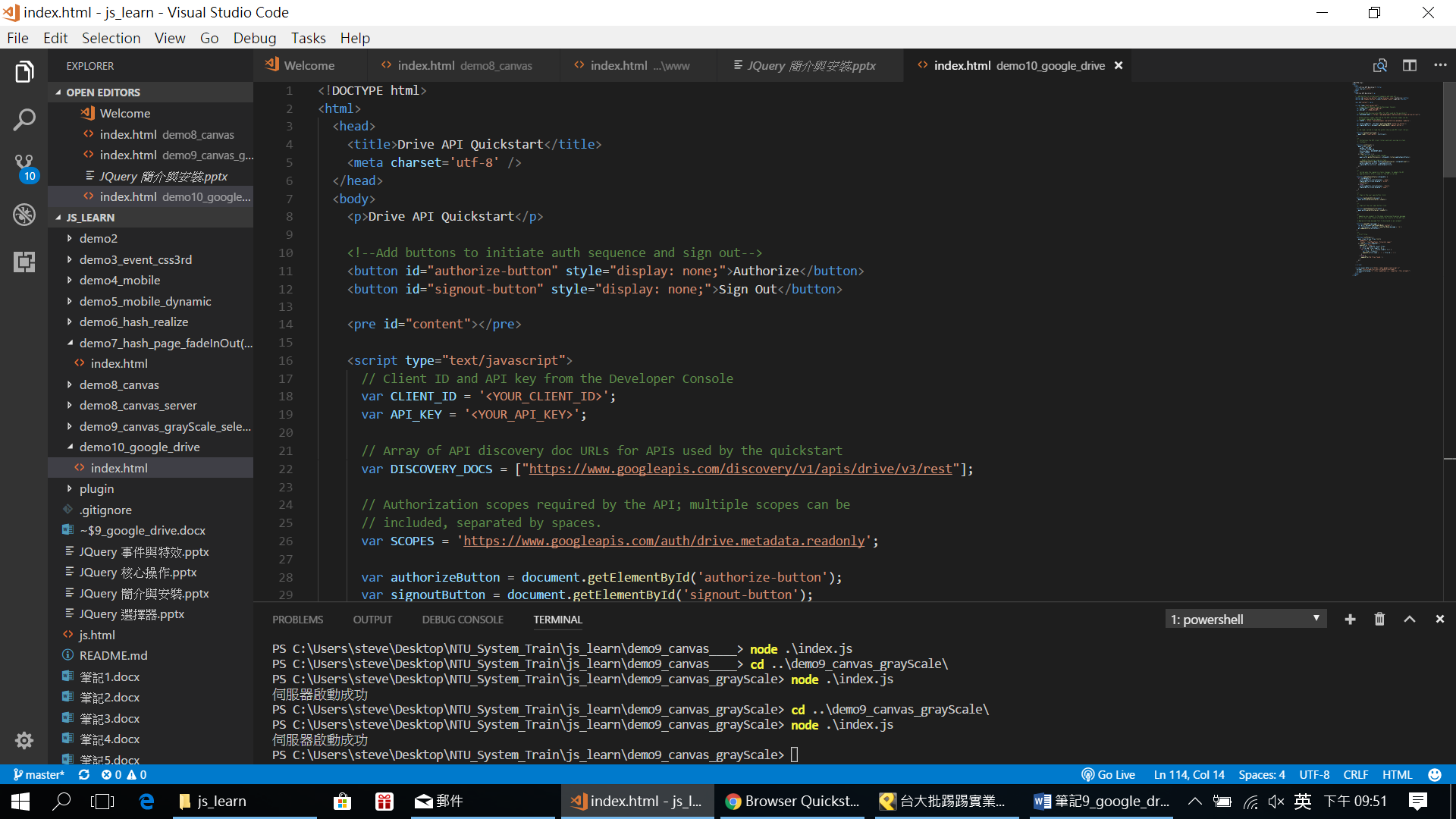
建立API Key:



16.

貼code:

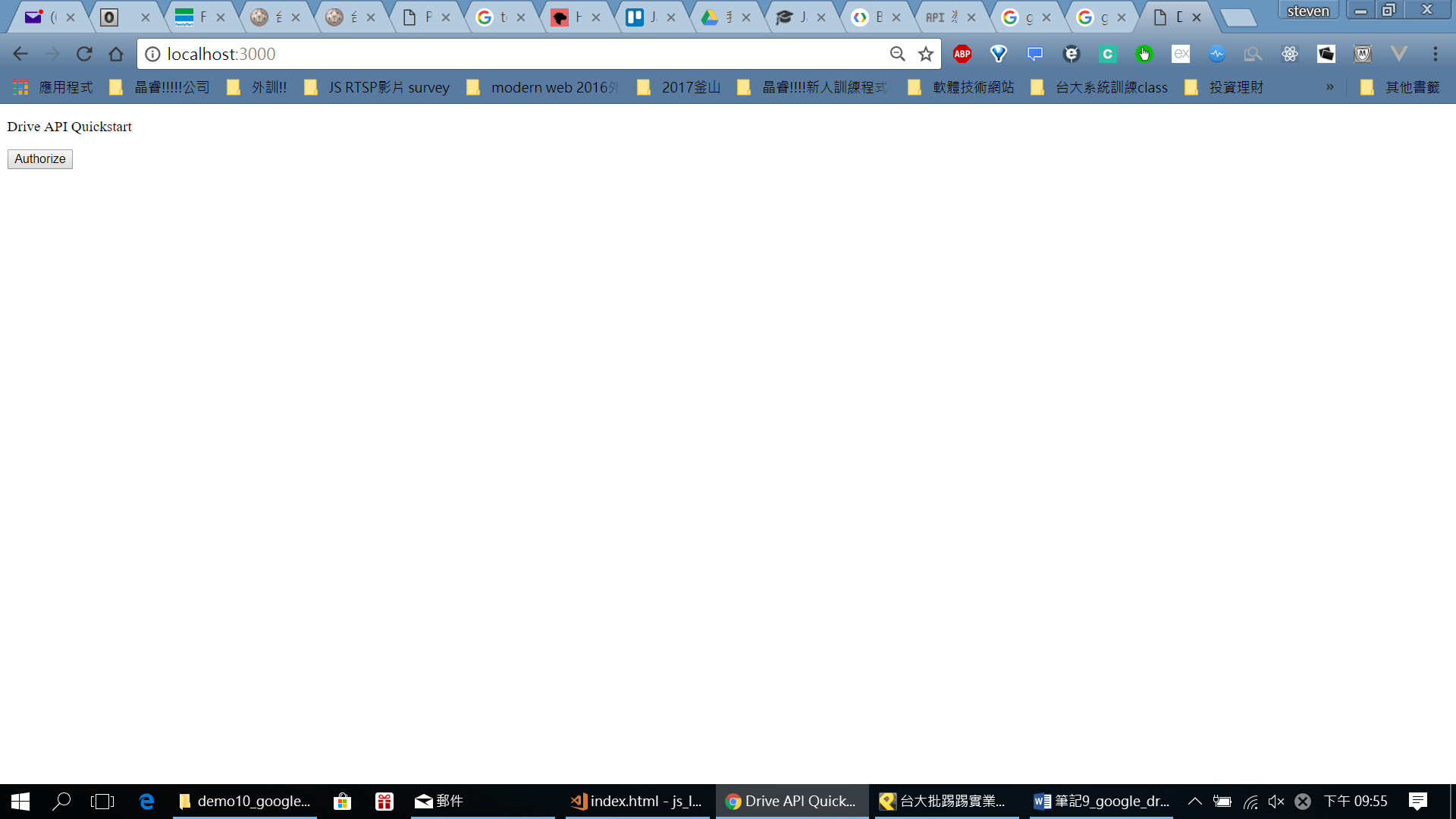


把client\_id (oauth)貼上

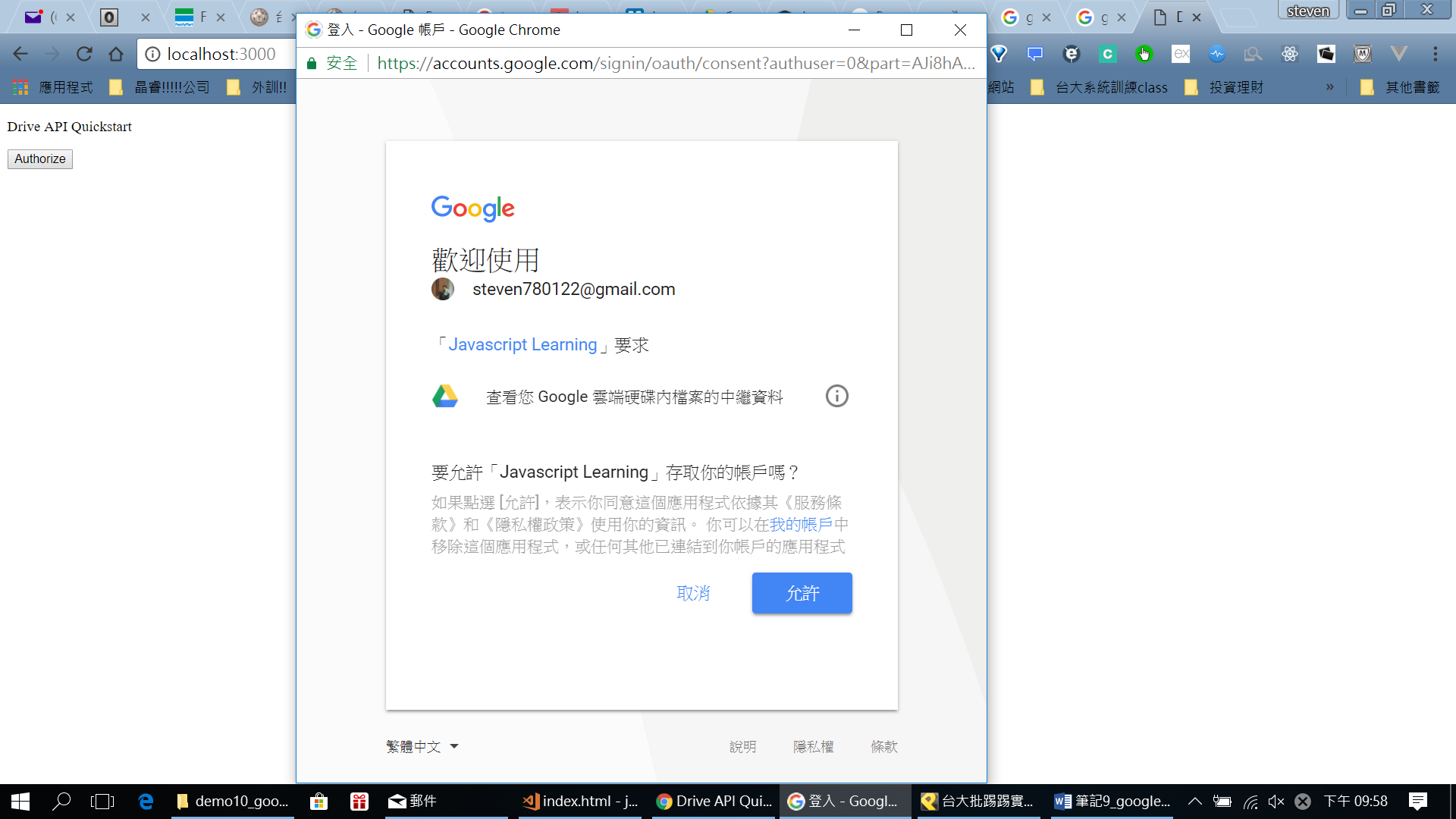
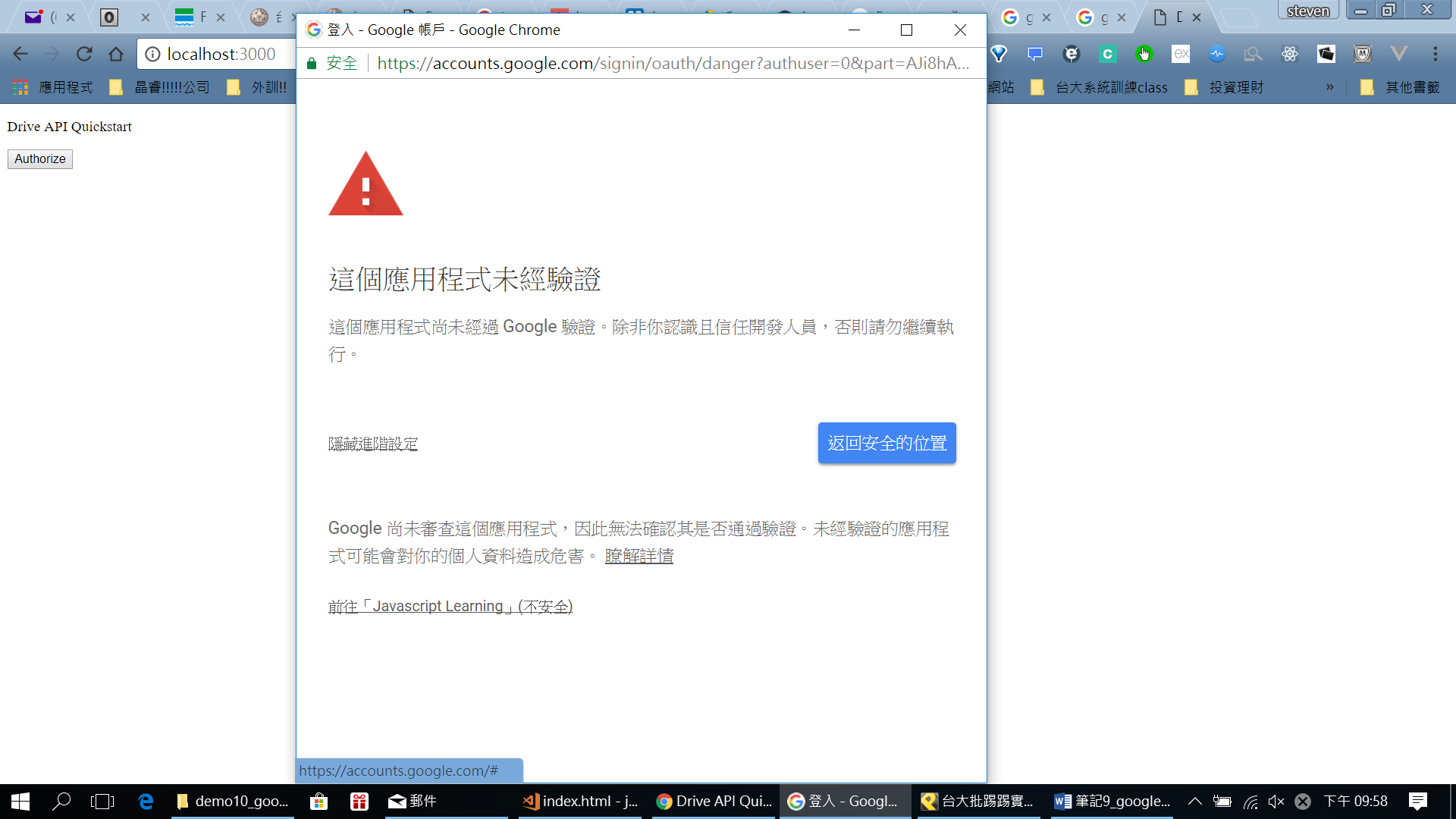
API key也是:

1. 因為我是用node (用localhost:3000)

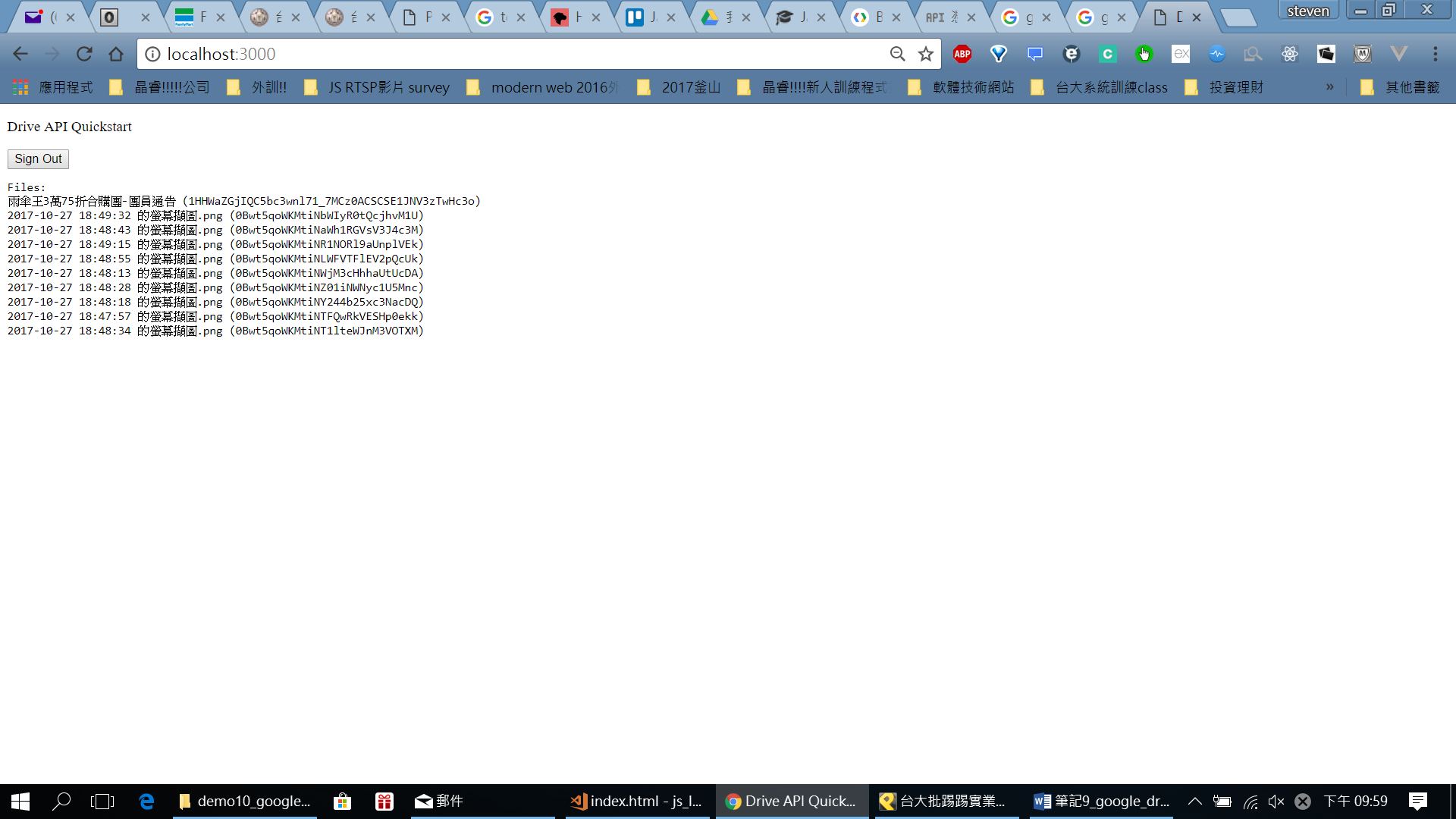
用node index.js後



把授權給剛設定的localhost存取



就抓到google drive最近修改過的title id



很重要!!!

注意各大網站很多都是用OAuth!!

該程序可以串各個Google服務

(<https://console.cloud.google.com/apis/library?project=javascript-learn&hl=zh-TW>)

這些服務都可以用自己的APP去串!!! Youtube, pagespeed等等

都是商機啊~!!!!!!!!!!!

>>>>>>>>>>>>>

下一步:

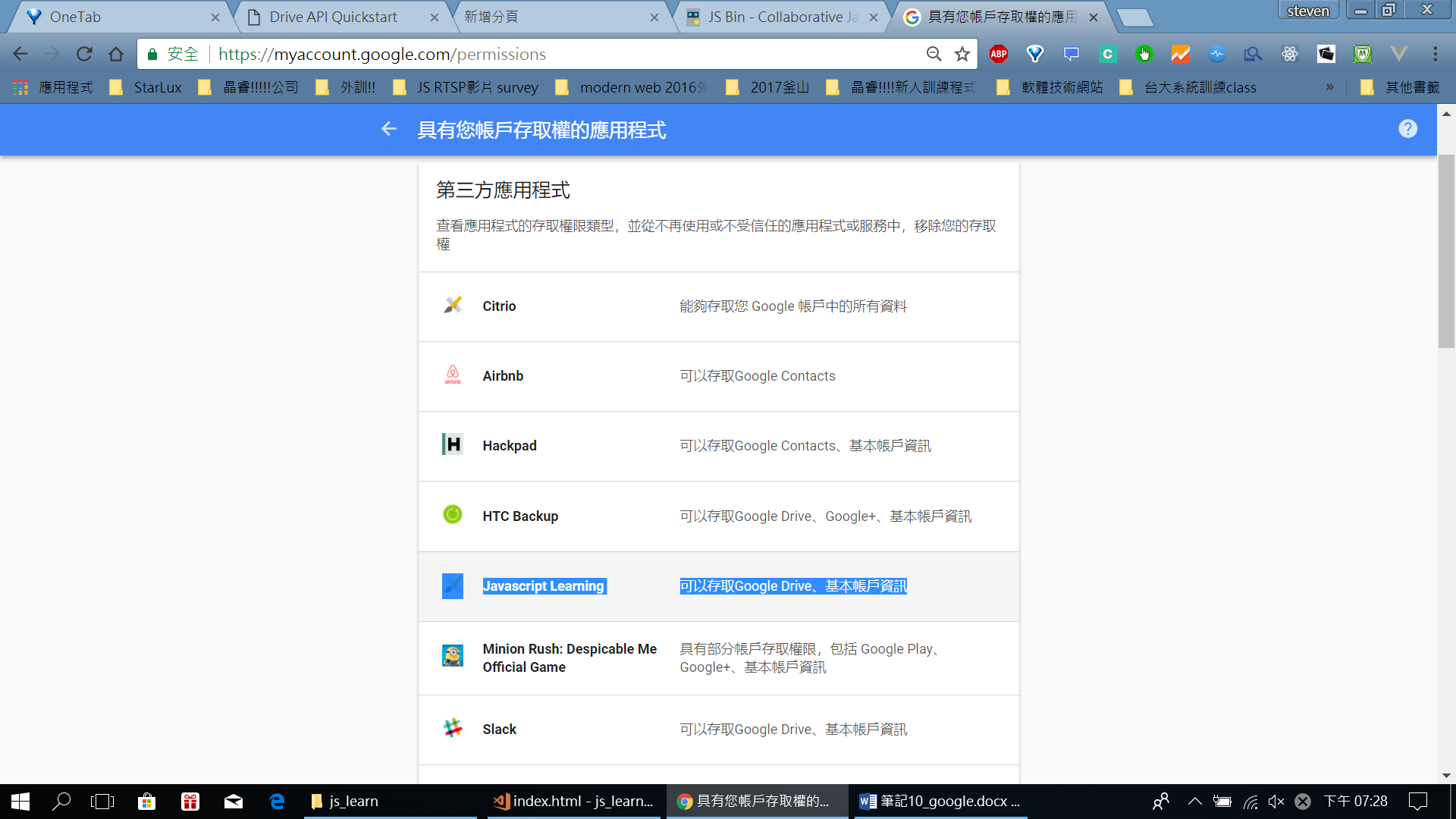
搜尋google 帳戶:

<https://myaccount.google.com/>

點:



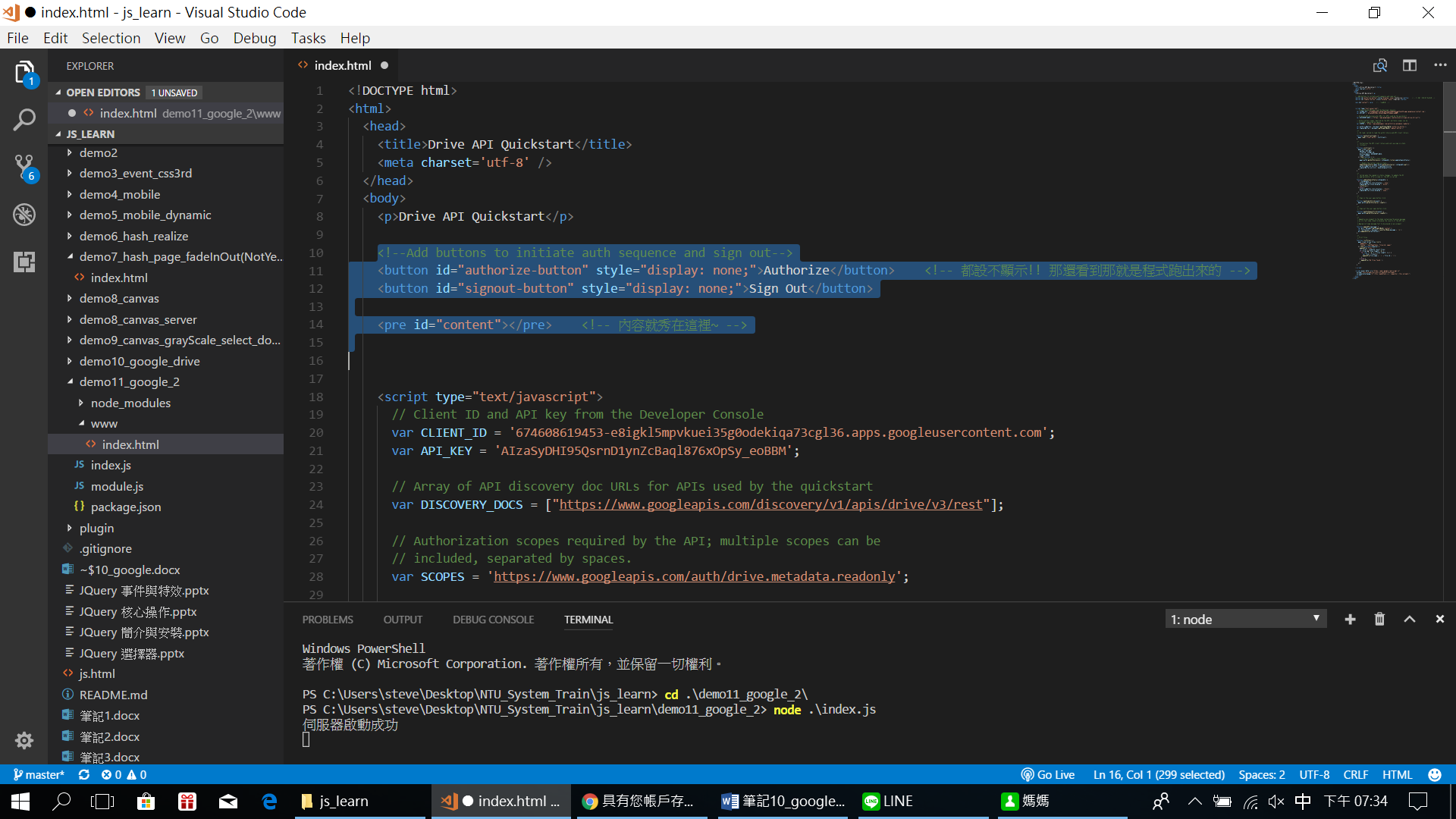
可以看到自己>>



如果在這裡編輯，移除授權~ 回到剛剛頁面可以看到需要再重新授權

回來看程式碼吧:

使用者進入網站，做初始化



幫我們多包一層:

Javascript Client Library (可google// 算是套件的子套件 ，因為太龐大了>> initClient

/\*\*

\* On load, called to load the auth2 library and API client library.

\*/

function handleClientLoad() {

gapi.load('client:auth2', initClient);

}

):

// 用來做偵測授權!，若授權就鼓勵他授權!若已授權就進主功能

/\*\*

\* Initializes the API client library and sets up sign-in state

\* listeners.

\*/

function initClient() {

gapi.client.init({

apiKey: API\_KEY,

clientId: CLIENT\_ID,

discoveryDocs: DISCOVERY\_DOCS,

scope: SCOPES

}).then(function () {

// Listen for sign-in state changes.

gapi.auth2.getAuthInstance().isSignedIn.listen(updateSigninStatus);

// Handle the initial sign-in state.

updateSigninStatus(gapi.auth2.getAuthInstance().isSignedIn.get());

authorizeButton.onclick = handleAuthClick;

signoutButton.onclick = handleSignoutClick;

});

}

/\*\*

\* Called when the signed in status changes, to update the UI

\* appropriately. After a sign-in, the API is called.

\*/

function updateSigninStatus(isSignedIn) {

if (isSignedIn) { // 使用者已授權，秀出登出按鈕，並且取得資料

authorizeButton.style.display = 'none';

signoutButton.style.display = 'block';

listFiles();

} else { // 使用者未授權，秀出授權按鈕!

authorizeButton.style.display = 'block';

signoutButton.style.display = 'none';

}

