HW6網頁設計作業1

網站上的定義是:跨來源資源共用（Cross-Origin Resource Sharing (CORS)）是一種使用額外 HTTP 標頭令目前瀏覽網站的使用者代理取得存取其他來源（網域）伺服器特定資源權限的機制。當使用者代理請求一個不是目前文件來源——例如來自於不同網域、通訊協定或通訊埠的資源時，會建立一個跨來源 HTTP 請求。

網站中有舉出一個例子，在講解現今網路上許多頁面所載入的資源，如 CSS 樣式表、圖片影像、以及指令碼（script）都來自與所在位置分離的網域，如內容傳遞網路（content delivery networks, CDN）。基於安全性考量，程式碼所發出的跨來源 HTTP 請求會受到限制。例如:XMLHttpRequest 及 Fetch 都遵守同源政策（same-origin policy）。

同源政策是指同源政策是網站安全的基礎。 https://domain-a.com 只能存取自己網站裡的資源 (圖片、影片、程式碼等)，不允許網站 https://domain-b.com 來存取。想要存取跨來源資源必須在某些特定情況下才被允許。

這代表網路應用程式所使用的 API 除非使用 CORS 標頭，否則只能請求與應用程式相同網域的 HTTP 資源。總結是跨來源資源共用（Cross-Origin Resource Sharing，簡稱 CORS）機制提供了網頁伺服器跨網域的存取控制，增加跨網域資料傳輸的安全性。現代瀏覽器支援在 API 容器中使用 CORS 以降低跨來源 HTTP 請求的風險。

在瀏覽器上，如果你想拿到一個網站的完整內容（可以完整讀取），基本上就只能透過 XMLHttpRequest 或是 fetch。若是這些跨來源的 AJAX 沒有限制的話，你就可以透過使用者的瀏覽器，拿到「任意網站」的內容，包含了各種可能有敏感資訊的網站。

因此瀏覽器會擋跨來源的 AJAX 是十分合理的一件事，就是為了安全性。

當一個支援 CORS 瀏覽器在網頁送出一個 request 時，會做下面的動作：

1. 瀏覽器根據送出 request 的 HTTP verb 與 header，判斷這個 request 是一個簡單請求或是非簡單請求。如果是一個簡單請求，則直接送出 request。
2. 如果是一個非簡單請求的 request，則進行 CORS preflight。先對伺服器送出一個 verb 為 OPTION 的 preflight request，它會帶有特定的 header 告訴伺服器接下來的 request 需要哪些跨網域連線的權限。

以上是2種不同的動作，這些說明讓我更了解不同的REQUEST會有不同的相對動作，看完許多網站介紹CORS之後，我覺得這個觀念對於我們正在學習網頁的學生來說非常重要，在不遠的將來我們就會遇到此問題了，因此現在趕快學習此觀念非常關鍵。