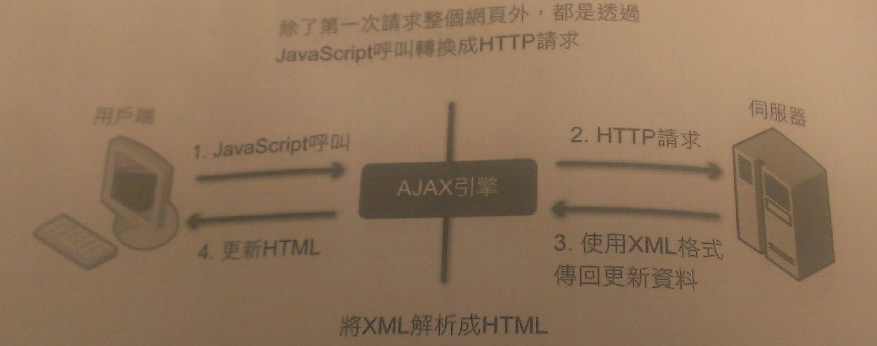
**AJAX的應用**

AJAX是Asynchronous Javascript and XML的縮寫，意思是非同步的Javascript和XML。就是使用XMLHttpRequest物件執行Http請求。

AJAX使用HTTP來與伺服器交換資料，不須從伺服器重新下載整個網頁。交換的資料量交換量並不大，而且瀏覽器不必解析與顯示整個網頁。



**AJAX組成**

1. Javascript:用來建立用戶端的AJAX函數，函數中使用DOM處理html部分網頁。
2. XMLHttpRequest物件:讓JavaScript程式以非同步方式，在背景模式中存取伺服器上的資料。
3. PHP:在伺服器上處理用戶端的JavaScript請求。

**AJAX應用問題**

1. 傳統網頁使用者按下連結，遊覽器會跳動畫面讓使用者知道正在進行連結，如果是使用XMLHttpRequest物件送出連結的HTTP請求，這些動作都會在背景中進行。如果HTTP請求嚴重延遲，使用者會無法得知有沒有在執行連結
2. AJAX應用程式中使用HTTP下載與顯示新的網頁時，URL位址不會改變，有時使用者會因為AJAX架構而無法直接進入網站中的某個網頁。
3. 瀏覽器的歷程記錄不會登錄XMLHttpRequest所用過的網頁，所以使用者無法用上下一頁功能，來找尋瀏覽器顯示過的網頁。
4. Ajax最主要缺點是使用者可以將瀏覽器jacascript除能，這會讓ajax沒有作用。

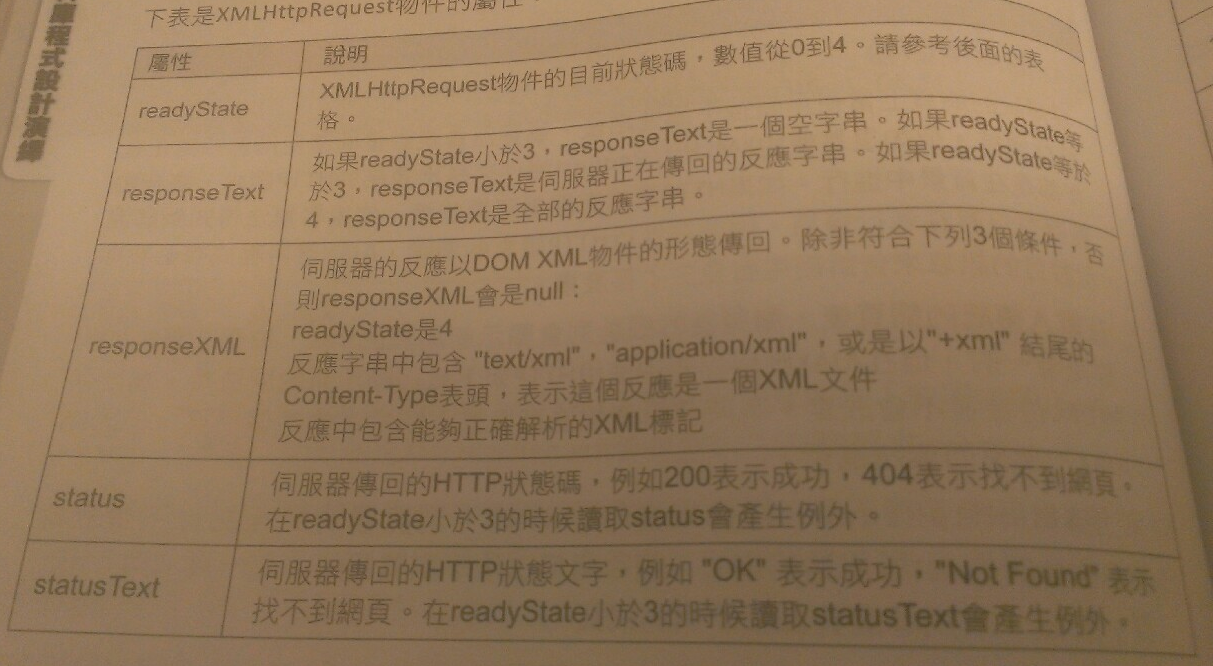
**XMLHttpRequest物件**

XMLHttpRequest物件讓JavaScript程式向伺服器執行非同步的HTTP請求，並接受伺服器的回應後，以背景模式更新部分網頁。

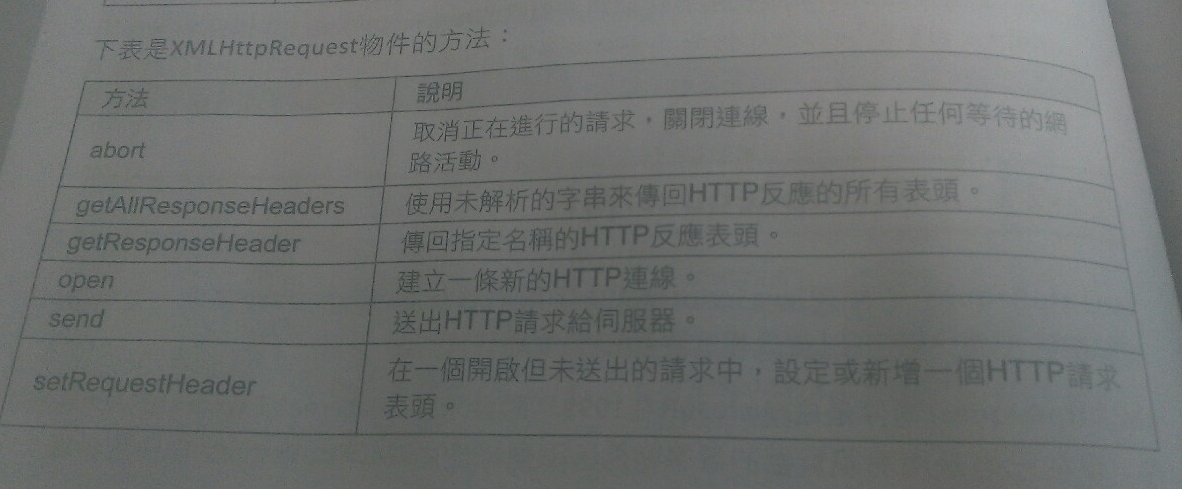
使用XMLHttpRequest物件步驟:

1. 建立一個XMLHttpRequest物件
2. 向伺服器執行非同步的呼叫，並定義一個callback函數 (在onreadystatechange方法做處理函數)。當伺服器傳回反應時，callback函數會自動呼叫
3. 在callback函數中解析伺服器的反應。
4. 在接收到伺服器傳回的資料執行網頁的更新。

XMLHttpRequest物件屬性如下圖:



XMLHttpRequest物件方法如下:



**建立XMLHttpRequest物件**

使用XMLHttpRequest物件的建構子



向伺服器提出HTTP請求

建立好XMLHttpRequest物件後，再來就是像伺服器提出HTTP請求，這個動作包含下列步驟:

1. 使用XMLHttpRequest物件的open方法開啟HTTP連線。
2. 如果要指定使用非同步的HTTP請求，設定onreadystatechange事件處理常式接收請求過程中的通知。
3. 使用setRequestHeader方法指定適當的HTTP請求表頭。
4. 使用send方法送出HTTP請求。

**讀取伺服器的反應**

1. 當接收伺服器反應，readystate屬性改變時，使用XMLHttpRequest物件的onreadystatechange事件處理常式做觸發動作。
2. responseText屬性傳回文字純文字或responseXML屬性傳回XML檔案的document物件。

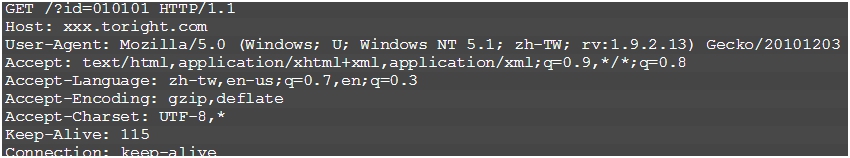
**補充:**

**為何要用到setRequestHeader (X) , 了解GET & POST**

**Get**

我們送出一個 GET 表單時，瀏覽器會自動將表單內容轉為 Query String 加在 URL 進行連線。

HTTP Request 封包的內容：

****

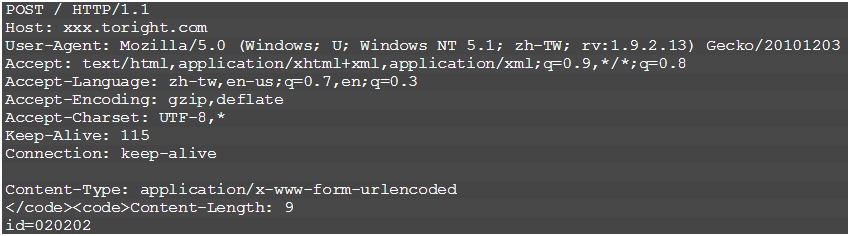
GET方式是沒有提交內容的,GET傳參數的方式就是通過虛擬地址傳送。

當action為get時候，瀏覽器用x-www-form-urlencoded的編碼方式把form數據轉換成一個字串（name1=value1&name2=value2...），然後把這個字串append到url後面，用?分割，加載這個新的url。

**POST**

 POST 是將表單資料放在 message-body 進行傳送

HTTP Request 封包的內容：



當action為post時候，瀏覽器把form數據封裝到http body中，然後發送到server。

**Form的enctype屬性為編碼方式**

常用有兩種：application/x-www-form-urlencoded和multipart/form-data，預設為application/x-www-form-urlencoded。

**application/x-www-form-urlencoded**

編碼方式把form數據轉換成一個字串（name1=value1&name2=value2...）

**multipart/form-data**

在傳送檔案(上傳檔案)的時候會使用到 multi-part 編碼，將檔案與其他的表單欄位一併放在 message-body 中進行傳送。

瀏覽器會把整個表單以控件為單位分割，並為每個部分加上Content-Disposition(form-data或者file),Content-Type(默認為text/plain),name(控件name)等信息，並加上分割符(boundary)。

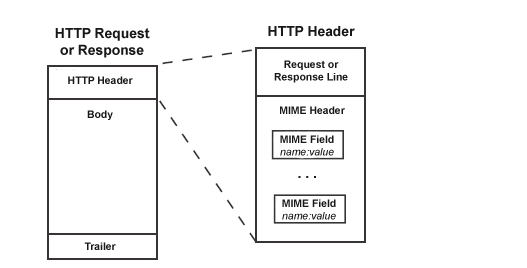
**為何要用到setRequestHeader**

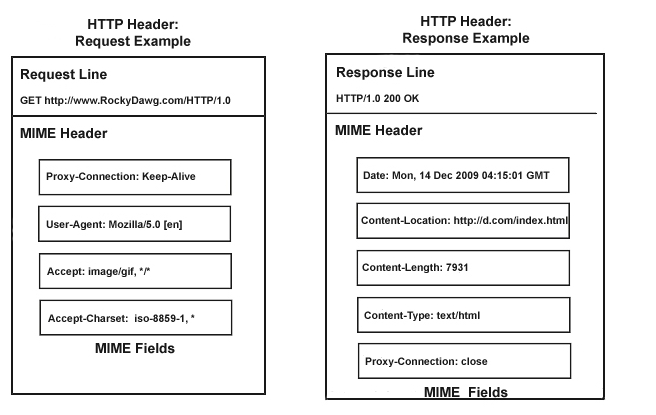
通常在HTTP協議裡，客戶端像服務器取得某個網頁的時候，必需送一個HTTP協議的header文件，告訴服務器客戶端要下載什麼信息以及相關的參數。

而XMLHTTP就是通過HTTP協議取得網站上的文件數據的，所以也要發送HTTP header給服務器。但是XMLHTTP預設的情況下有些參數(如content-type)可能沒有說明在HTTP的header里，這時當我們需要修改或添加這些參數時就用到了setRequestHeader方法。

所以當使用POST方法做AJAX時，需使用setRequestHeader為Header添加application/x-www-form-urlencoded 參數。

**HTTP Request/Response and Header Structure**

****



參考:

<http://www.ourys.com/post/110.html>

<https://www.imququ.com/post/four-ways-to-post-data-in-http.html>

<http://www.w3school.com.cn/ajax/ajax_xmlhttprequest_send.asp>