開發網頁時，有一個很重要的觀念，那就是HTTP協定是無狀態的。「簡單來說就是伺服器處理完你的request之後就與你無關了，而且它也不會記得你和它之間的故事，你們的關係形同陌生人，殘忍的是你又看著它與其他人展開新的故事，寫code再苦，也比不上它在你心中留下的傷痛，恨自己到底為什麼，直到現在還守著這份回憶，好似期待有一天還能夠回到當初，好想再看到那個屬於自己的微笑，只要一次就好，我真的...真的好想妳。」

HTTP無狀態 特性，伺服器並不會知道你之前做了什麼，常見的例子還有購物車，買了好幾個商品要結帳，結果伺服器不知道我到底買了什麼。cookie可以用來解決這個問題，它能夠儲存一些資訊，像是我拿著會員卡，就能夠直接進入商店，因為上面的卡號說明了我是這家店已經付費過的客人，也就是他們的會員，所以我不用再付費就能夠入場。cookie的特性，1.可以紀錄使用者訊息。2.儲存在客戶端。3.連線時會自動帶上，但過多的cookie可能會浪費流量、或是帶上無用之cookie。4.大小限制4kb左右。5.能夠設置過期時間。6.專屬於某網域(路徑)，也就是google.com的頁面不能存取facebook.com的cookie。cookie 說穿了就是 key=value 的形式而已，透過document.cookie查看時，發現它們全部被組成字串，如果只是要單純存取某個屬性的值而已，不就還要切割字串嗎?JS本身沒有提供這些方法，但W3C有附上常用操作的函數。

取得指定屬性的值

function getCookie(cname) {

var name = cname + "=";

var decodedCookie = decodeURIComponent(document.cookie);

var ca = decodedCookie.split(';');

for(var i = 0; i <ca.length; i++) {

var c = ca[i];

while (c.charAt(0) == ' ') {

c = c.substring(1);

}

if (c.indexOf(name) == 0) {

return c.substring(name.length, c.length);

}

}

return "";

}

設置 cookie

function setCookie(cname, cvalue, exdays) {

var d = new Date();

d.setTime(d.getTime() + (exdays\*24\*60\*60\*1000));

var expires = "expires="+ d.toUTCString();

document.cookie = cname + "=" + cvalue + ";" + expires + ";path=/";

}

需要注意的是，該path屬性不能防止未經授權從不同路徑讀取cookie。使用DOM可以輕鬆繞過它，例如通過使用<iframe> cookie的路徑創建隱藏元素，然後訪問此iframe的contentDocument.cookie屬性。由於相同的源策略，保護cookie的唯一方法是使用不同的域或子域。Cookies通常在Web應用程序中用於識別用戶及其經過身份驗證的會話。從Web應用程序中竊取cookie會導致劫持經過身份驗證的用戶的會話。竊取cookie的常見方法包括使用社交工程或利用應用程序中的跨站腳本(XSS)漏洞，另外cookie屬性可以通過HTTPOnly阻止通過 Javascript 訪問cookie值來幫助減輕這種攻擊。