

研讀 http cookie 及其用法

● cookie

1. 因為 HTTP 的無狀態性，伺服器並不知道使用者在每個頁面跳轉時到底帶入了什麼資訊，而 Cookie 就是用來繞開 HTTP 的無狀態性的「額外手段」之一，讓伺服器可以設定或讀取 Cookies 中所包含資訊，藉此維護使用者在使用服務時，在背景完成並可以持續跟伺服器發送請求以及對談中的狀態。
2. Cookie 是一個「小型文字檔案」，一個 Cookie 大約只能儲存 4 MB 的資料，Cookie 是一個未經過加密的純文字檔案，為伺服器（Server）傳送給瀏覽器（Client）的一小片段資料，並請瀏覽器保存起來，以便往後向相同的伺服器發送請求時，附上這 Cookie 的資料
3. cookie 是儲存在我們用戶的本機端，通常可以儲存的地方有兩個：記憶體或硬體內。
 - (1) 記憶體是由 Browser(瀏覽器)來維護的，通常會在瀏覽器關閉後清除，而各個瀏覽器之間的 Cookie 是無法相互使用，也就是說對於在同一台電腦上使用 Chrome 或是 Firefox，儘管操作的是同一個人，卻是會認成兩個不一樣的角色
 - (2) 硬體的 Cookie 則會有一個保存期限，除非過期或是手動刪除，不然他的儲存時間會較瀏覽器來的長。
4. 分類：
 - (1) 完全必要的 Cookie：維持網站運作所必需的 Cookie。例如，使您能登入到網站的安全區域、使用購物車或使用電子帳單服務的 Cookie。
 - (2) 分析/性能 Cookie：這類 Cookie 使網站能夠識別訪客身份，計算訪客數量，瞭解訪客使用網站時查看了哪些內容。這有助於透過確保使用者輕鬆地找到其正在尋找的內容等方法，改進我們網站的運作方式。
 - (3) 功能 Cookie：這些 Cookie 用於在您返回我們的網站時識別您的身份，藉此使我們能夠為您提供客製化內容，例如：透過稱呼您的名字來問候、記住偏好。
 - (4) 目標 Cookie：這些 Cookie 會記錄對網站的存取、瀏覽過的網頁和追蹤過的連結。使用這些資訊使網站及其上面顯示的廣告更切合您的興趣。基於此目的，我們還可能會與協力廠商分享這些資訊（以匿名形式）。

5. 流程：

- (1) 使用者透過瀏覽器登入畫面登入
- (2) 伺服器端通過驗證後，可以將使用者「已經登入」的資訊附在 **Cookie** 中回傳，並請瀏覽器保存起來
- (3) 往後，每當瀏覽器對伺服器發出請求時，會一併附上存有使用者「已經登入」狀態資訊的 **Cookie** 給伺服器
- (4) 伺服器透過 **Cookie** 就能辨識這位使用者已經通過驗證了

6. 常見用途：

因為 **HTTP** 協定是無狀態的，即伺服器不知道使用者上一次做了什麼，這嚴重阻礙了互動式 **Web** 應用程式的實現，**Cookie** 就是用來繞開 **HTTP** 的無狀態性的手段，伺服器可以設定或讀取 **Cookie** 中包含的資訊，藉此維護使用者跟伺服器對談中的狀態

- (1) 儲存和追蹤使用者行為
- (2) 儲存用戶登入、購物車等伺服器所需的資訊
- (3) 儲存使用者設定和偏好等
- (4)

7. 依網域所有權分為第一方 **cookie** 與第三方 **cookie**

- (1) 接受第一方 **Cookie**，授權給信任網站，不僅能提升瀏覽體驗，對於個資也有所保障
- (2) 第三方 **Cookie** 會較難以掌握來源網站的安全性

◆ 第一方 **cookie**

1. 由自己所瀏覽的網站建立
2. 記錄使用者的資訊及登錄狀況
3. 通常會在你剛進入網站時詢問是否同意使用 **Cookie**，若不同意可能會讓網站無法正常運作

◆ 第三方 **cookie**

1. 由造訪的第一方網站上的其他網站廣告之網站所建立
2. 若點擊網頁中出現的廣告連結，伺服器就會收到第三方網站的 **Cookie**
3. 安全性較差

例子一：

index.jsp

```
1  <%@ page language="java" contentType="text/html; charset=utf-8"
2      pageEncoding="utf-8"%>
3      <%@ page import="java.net.URLDecoder" %>
4  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
5  <html>
6  <head>
7  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
8  <title>取出Cookie</title>
9  </head>
10 <body>
11 <%! String name=""; %>
12 <%
13     Cookie[] cookies=request.getCookies();
14     if(cookies!=null){
15         for(Cookie c:cookies){
16             if(c.getName().equals("name")){
17                 name=URLDecoder.decode(c.getValue(),"utf-8");//对字符串进行解码,类似于字符串的解码
18             }
19         }
20     }
21 %>
22 <form name="form" action="deal.jsp" method="post">
23     <%
24         if("".equals(name)){
25             %>
26             <strong>欢迎初次登录</strong>
27         <%
28             }else{
29                 %>
30                 <strong>欢迎再次登录,已保存上次登录用户名</strong>
31             <%> %><br>
32             请输入姓名: <input type="text" name="name" value="<%=name%%">
33             <input type="submit" value="提交">
34         </form>
35     </body>
36 </html>
```

deal.jsp:

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <%@ page import="java.net.URLEncoder" %>
4 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
5 <html>
6 <head>
7 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
8 <title>存储Cookie</title>
9 </head>
10 <body>
11 <%
12     request.setCharacterEncoding("utf-8");//对request传过来的参数设置编码格式,以免传入中文的时候出现问题,必须在request.getPar
13     String name=URLEncoder.encode(request.getParameter("name"),"utf-8");//对字符串以utf8的格式进行编码
14     Cookie cookie=new Cookie("name",name);
15     response.addCookie(cookie);
16 %>
17 <script type="text/javascript">window.location.href="index.jsp"</script><!-- js中实现页面的跳转 -->
18 </body>
19 </html>
```

运行结果:

刚开始的界面,我们下面输入“张三”



这是运行成功后的结果。



例子二：

使用 JavaScript，可以像這樣創建 cookie：

```
document.cookie = "username=John Doe";
```

您還可以添加到期日期（以 UTC 時間表示）。默認情況下，關閉瀏覽器時會刪除 cookie：

```
document.cookie = "username=John Doe; expires=Thu, 18 Dec 2013 12:00:00 UTC";
```

使用路徑參數，您可以告訴瀏覽器 cookie 屬於哪個路徑。默認情況下，cookie 屬於當前頁面。

```
document.cookie = "username=John Doe; expires=Thu, 18 Dec 2013 12:00:00 UTC; path=/";
```

使用 JavaScript 讀取 Cookie

使用 JavaScript，cookie 可以這樣讀取：

```
let x = document.cookie;
```

`document.cookie` 將在一個字符串中返回所有 cookie，就像：`cookie1=value; cookie2=值; cookie3=值;`

使用 JavaScript 更改 Cookie

使用 JavaScript，您可以像創建 cookie 一樣更改 cookie：

```
document.cookie = "username=John Smith; expires=Thu, 18 Dec 2013 12:00:00 UTC; path=/";
```

```
function setCookie(cname, cvalue, exdays) {
    const d = new Date();
    d.setTime(d.getTime() + (exdays * 24 * 60 * 60 * 1000));
    let expires = "expires="+d.toUTCString();
    document.cookie = cname + "=" + cvalue + ";" + expires + ";path=/";
}

function getCookie(cname) {
    let name = cname + "=";
    let ca = document.cookie.split(';');
    for(let i = 0; i < ca.length; i++) {
        let c = ca[i];
        while (c.charAt(0) == ' ') {
            c = c.substring(1);
        }
        if (c.indexOf(name) == 0) {
            return c.substring(name.length, c.length);
        }
    }
    return "";
}

function checkCookie() {
    let user = getCookie("username");
    if (user != "") {
        alert("Welcome again " + user);
    } else {
        user = prompt("Please enter your name:", "");
        if (user != "" && user != null) {
            setCookie("username", user, 365);
        }
    }
}
```