# 8.10 Modifying Cookie Values: Tracking User Access Counts

In the previous examples, we sent a cookie to the user only on the first visit. Once the cookie had a value, we never changed it. This approach of a single cookie value is surprisingly common since cookies frequently contain nothing but unique user identifiers: all the real user data is stored in a database—the user identifier is merely the database key.

But what if you want to periodically change the value of a cookie? How do you do so?

- To *replace* a previous cookie value, send the same cookie name with a different cookie value. If you actually use the incoming `Cookie` objects, don't forget to do `response.addCookie`; merely calling `setValue` is not sufficient. You also need to reapply any relevant cookie attributes by calling `setMaxAge`, `setPath`, etc.—cookie attributes are not specified for incoming cookies. Reapplying these attributes means that reusing the incoming `Cookie` objects saves you little, so many developers don't bother.

- To instruct the browser to *delete* a cookie, use `setMaxAge` to assign a maximum age of 0.

Listing 8.6 presents a servlet that keeps track of how many times each client has visited the page. It does this by making a cookie whose name is `accessCount` and whose value is the actual count. To accomplish this task, the servlet needs to repeatedly replace the cookie value by resending a cookie with the identical name.

Figure 8-10 shows some typical results.

## Listing 8.6 ClientAccessCounts.java

```
package coreservlets;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

/** Servlet that prints per-client access counts. */

public class ClientAccessCounts extends HttpServlet {
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    String countString =
      CookieUtilities.getCookieValue(request,
                                     "accessCount",
                                     "1");
    int count = 1;
    try {
      count = Integer.parseInt(countString);
    } catch(NumberFormatException nfe) { }
    LongLivedCookie c =
      new LongLivedCookie("accessCount",
                          String.valueOf(count+1));
    response.addCookie(c);
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String title = "Access Count Servlet";
    String docType =
```

```
      "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
      "Transitional//EN\">\n";
    out.println(docType +
                "<HTML>\n" +
                "<HEAD><TITLE>" + title + "</TITLE></HEAD>\n" +
                "<BODY BGCOLOR=\"#FDF5E6\">\n" +
                "<CENTER>\n" +
                "<H1>" + title + "</H1>\n" +
                "<H2>This is visit number " +
                count + " by this browser.</H2>\n" +
                "</CENTER></BODY></HTML>");
  }
}
```

**Figure 8-10. Users each see their own access count. Also, Internet Explorer and Netscape maintain cookies separately, so the same user sees independent access counts with the two browsers.**