



SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107



AN AUTONOMOUS INSTITUTION

Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIT 3-SERVER SIDE PROGRAMMING

3.4 UNDERSTANDING COOKIES

- Cookies are text files stored on the client computer and they are kept for various information tracking purpose. Java Servlets transparently supports HTTP cookies.
- There are three steps involved in identifying returning users –
 - Server script sends a set of cookies to the browser. For example name, age, or identification number etc.
 - Browser stores this information on local machine for future use.
 - When next time browser sends any request to web server then it sends those cookies information to the server and server uses that information to identify the user.

3.4.1 The Anatomy of a Cookie

- Cookies are usually set in an HTTP header (although JavaScript can also set a cookie directly on a browser). A servlet that sets a cookie might send headers that look something like this –

HTTP/1.1 200 OK

Date: Fri, 04 Feb 2000 21:03:38 GMT

Server: Apache/1.3.9 (UNIX) PHP/4.0b3

Set-Cookie: name = xyz; expires = Friday, 04-Feb-07 22:03:38 GMT;

path = /; domain = tutorialspoint.com

Connection: close

Content-Type: text/html

3.5.4 Setting Cookies with Servlet

- Setting cookies with servlet involves three steps –
 - i. Creating a Cookie object – You call the Cookie constructor with a cookie name and a cookie value, both of which are strings.

Cookie cookie = new Cookie("key","value");

- ii. Keep in mind, neither the name nor the value should contain white space or any of the following characters –

[] () = , " / ? @ : ;

- iii. Setting the maximum age – You use `setMaxAge` to specify how long (in seconds) the cookie should be valid. Following would set up a cookie for 24 hours.

`cookie.setMaxAge(60 * 60 * 24);`

- iv. Sending the Cookie into the HTTP response headers – You use `response.addCookie` to add cookies in the HTTP response header as follows –

`response.addCookie(cookie);`

Example

```
// Import required java libraries
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

// Extend HttpServlet class
public class HelloForm extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // Create cookies for first and last names.
        Cookie firstName = new Cookie("first_name", request.getParameter("first_name"));
        Cookie lastName = new Cookie("last_name", request.getParameter("last_name"));
        // Set expiry date after 24 Hrs for both the cookies.
        firstName.setMaxAge(60*60*24);
        lastName.setMaxAge(60*60*24);
        // Add both the cookies in the response header.
        response.addCookie( firstName );
        response.addCookie( lastName );
        // Set response content type
        response.setContentType("text/html");
```

```

PrintWriter out = response.getWriter();

String title = "Setting Cookies Example";

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 = \"#f0f0f0\">\n" +

"<h1 align = \"center\">" + title + "</h1>\n" +

"<ul>\n" +

"  <li><b>First Name</b>: "

+ request.getParameter("first_name") + "\n" +

"  <li><b>Last Name</b>: "

+ request.getParameter("last_name") + "\n" +

"</ul>\n" +

"</body>

</html>"); }}

```

3.5.3 Servlet Cookies Methods

| Sr.No. | Method & Description |
|--------|---|
| 1 | public void setDomain(String pattern) This method sets the domain to which cookie applies, for example tutorialspoint.com. |
| 2 | public String getDomain() This method gets the domain to which cookie applies, for example tutorialspoint.com. |
| 3 | public void setMaxAge(int expiry) This method sets how much time (in seconds) should elapse before the cookie expires. If you don't set this, the cookie will last only for the current session. |
| 4 | public int getMaxAge() This method returns the maximum age of the cookie, specified in seconds, By default, -1 indicating the cookie will persist until browser shutdown. |
| 5 | public String getName() |

| | |
|----|--|
| | This method returns the name of the cookie. The name cannot be changed after creation. |
| 6 | public void setValue(String newValue) This method sets the value associated with the cookie |
| 7 | public String getValue() This method gets the value associated with the cookie. |
| 8 | public void setPath(String uri) This method sets the path to which this cookie applies. If you don't specify a path, the cookie is returned for all URLs in the same directory as the current page as well as all subdirectories. |
| 9 | public String getPath() This method gets the path to which this cookie applies. |
| 10 | public void setSecure(boolean flag) This method sets the boolean value indicating whether the cookie should only be sent over encrypted (i.e. SSL) connections. |
| 11 | public void setComment(String purpose) This method specifies a comment that describes a cookie's purpose. The comment is useful if the browser presents the cookie to the user. |
| 12 | public String getComment() This method returns the comment describing the purpose of this cookie, or null if the cookie has no comment. |

3.5.4 Reading Cookies with Servlet

- To read cookies, you need to create an array of javax.servlet.http.Cookie objects by calling the getCookies() method of HttpServletRequest. Then cycle through the array, and use getName() and getValue() methods to access each cookie and associated value.

3.5.6 Delete Cookies with Servlet

- To delete cookies is very simple. If you want to delete a cookie then you simply need to follow up following three steps –
 - Read an already existing cookie and store it in Cookie object.
 - Set cookie age as zero using **setMaxAge()** method to delete an existing cookie
 - Add this cookie back into response header.