



## Session Tracking Lecture 6 – Chapter 9

Core Servlets & JSP book: [www.coreservlets.com](http://www.coreservlets.com)  
More Servlets & JSP book: [www.moreservlets.com](http://www.moreservlets.com)  
Servlet and JSP Training Courses: [courses.coreservlets.com](http://courses.coreservlets.com)

Slides © Marty Hall, <http://www.coreservlets.com>, book © Sun Microsystems Press

## Agenda

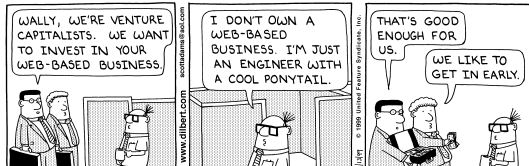
- The purpose of session tracking
- Rolling your own session tracking
- The session tracking API
- Per-client access counts
- Shopping carts and online stores

Session Tracking

[www.coreservlets.com](http://www.coreservlets.com)

## Session Tracking and E-Commerce

- **Why session tracking?**
  - When clients at on-line store add item to their shopping cart, how does server know what's already in cart?
  - When clients decide to proceed to checkout, how can server determine which previously created cart is theirs?



Dilbert used with permission of United Syndicates Inc.

Session Tracking

[www.coreservlets.com](http://www.coreservlets.com)

## Rolling Your Own Session Tracking: Cookies

- **Idea: associate cookie with data on server**

```
String sessionID = makeUniqueString();
Hashtable sessionInfo = new Hashtable();
Hashtable globalTable = findTableStoringSessions();
globalTable.put(sessionID, sessionInfo);
Cookie sessionCookie =
    new Cookie("JSESSIONID", sessionID);
sessionCookie.setPath("/");
response.addCookie(sessionCookie);
```
- **Still to be done:**
  - Extracting cookie that stores session identifier
  - Setting appropriate expiration time for cookie
  - Associating the hash tables with each request
  - Generating the unique session identifiers

Session Tracking

[www.coreservlets.com](http://www.coreservlets.com)

## Rolling Your Own Session Tracking: URL-Rewriting

- **Idea**
  - Client appends some extra data on the end of each URL that identifies the session
  - Server associates that identifier with data it has stored about that session
  - E.g., <http://host/path/file.html;jsessionid=1234>
- **Advantage**
  - Works even if cookies are disabled or unsupported
- **Disadvantages**
  - Lots of tedious processing
  - Must encode all URLs that refer to your own site
  - Links from other sites and bookmarks can fail

Session Tracking

[www.coreservlets.com](http://www.coreservlets.com)

## Rolling Your Own Session Tracking: Hidden Form Fields

- **Idea:**

```
<INPUT TYPE="HIDDEN" NAME="session" VALUE="...">
```
- **Advantage**
  - Works even if cookies are disabled or unsupported
- **Disadvantages**
  - Lots of tedious processing
  - All pages must be the result of form submissions

Session Tracking

[www.coreservlets.com](http://www.coreservlets.com)

## The Session Tracking API

- **Session objects live on the server**
- **Automatically associated with client via cookies or URL-rewriting**
  - Use `request.getSession(true)` to get either existing or new session
    - Behind the scenes, the system looks at cookie or URL extra info and sees if it matches the key to some previously stored session object. If so, it returns that object. If not, it creates a new one, assigns a cookie or URL info as its key, and returns that new session object.
- **Hashtable-like mechanism lets you store arbitrary objects inside session**
  - `setAttribute` (`putValue` in 2.1) stores values
  - `getAttribute` (`getValue` in 2.1) retrieves values

Session Tracking

www.coreservlets.com

## Accessing Session Data

```
HttpSession session = request.getSession(true);
ShoppingCart cart =
    (ShoppingCart)session.getAttribute("shoppingCart");
if (cart == null) { // No cart already in session
    cart = new ShoppingCart();
    session.setAttribute("shoppingCart", cart);
}
doSomethingWith(cart);
```

Session Tracking

www.coreservlets.com

## HttpSession Methods

- **getAttribute, getValue [2.1]**
  - Extracts a previously stored value from a session object. Returns null if no value is associated with given name.
- **setAttribute, putValue [2.1]**
  - Associates a value with a name. Monitor changes: values implement `HttpSessionBindingListener`.
- **removeAttribute, removeValue [2.1]**
  - Removes values associated with name.
- **getAttributeNames, getValueNames [2.1]**
  - Returns names of all attributes in the session.
- **getId**
  - Returns the unique identifier.

Session Tracking

www.coreservlets.com

## HttpSession Methods (Continued)

- **isNew**
  - Determines if session is new to *client* (not to *page*)
- **getCreationTime**
  - Returns time at which session was first created
- **getLastAccessedTime**
  - Returns time at which session was last sent from client
- **getMaxInactiveInterval, setMaxInactiveInterval**
  - Gets or sets the amount of time session should go without access before being invalidated
- **invalidate**
  - Invalidates the session and unbinds all objects associated with it

Session Tracking

www.coreservlets.com

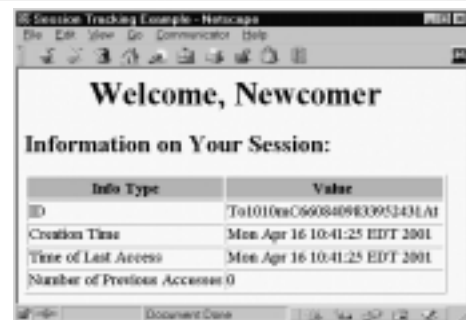
## A Servlet Showing Per-Client Access Counts

```
public void doGet(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String title = "Session Tracking Example";
    HttpSession session = request.getSession(true);
    String heading;
    Integer accessCount =
        (Integer)session.getAttribute("accessCount");
    if (accessCount == null) {
        accessCount = new Integer(0);
        heading = "Welcome, Newcomer";
    } else {
        heading = "Welcome Back";
        accessCount = new Integer(accessCount.intValue() + 1);
    }
    session.setAttribute("accessCount", accessCount);
}
```

Session Tracking

www.coreservlets.com

## First Visit to ShowSession Servlet



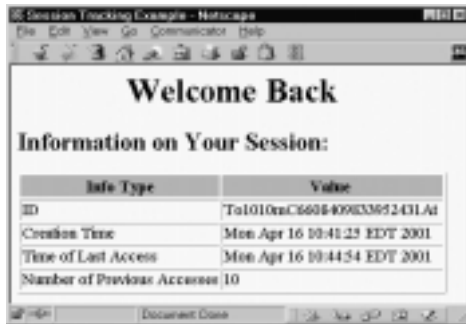
The screenshot shows a web browser window titled "Session Tracking Example - Netscape". The page content includes a heading "Welcome, Newcomer" and a section titled "Information on Your Session:". Below this is a table with session details.

Info Type	Value
ID	To1010eaC6608409E3995243A1
Creation Time	Mon Apr 16 10:41:25 EDT 2001
Time of Last Access	Mon Apr 16 10:41:25 EDT 2001
Number of Previous Accesses	0

Session Tracking

www.coreservlets.com

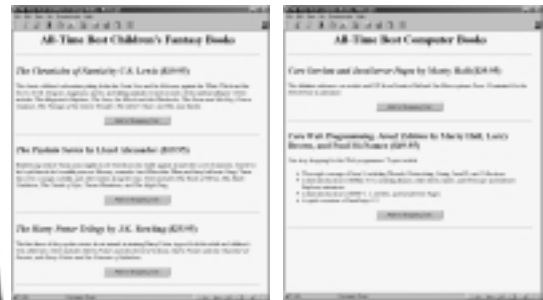
## Eleventh Visit to ShowSession Servlet



Session Tracking

www.coreservlets.com

## Session Tracking and Shopping Carts



Session Tracking

www.coreservlets.com

## Session Tracking and Shopping Carts (Continued)



Session Tracking

www.coreservlets.com

## Summary

- Although it usually uses cookies behind the scenes, the session tracking API is higher-level and easier to use than the cookie API
  - If server supports URL-rewriting, *your* code unchanged
- Session information lives on server
  - Cookie or extra URL info associates it with a user
- Obtaining session
  - request.getSession(true)
- Associating values with keys
  - session.setAttribute (or session.putValue)
- Finding values associated with keys
  - session.getAttribute (or session.getValue)
  - Always check if this value is null

Session Tracking

www.coreservlets.com



## Questions?

Core Servlets & JSP book: [www.coreservlets.com](http://www.coreservlets.com)  
 More Servlets & JSP book: [www.moreservlets.com](http://www.moreservlets.com)  
 Servlet and JSP Training Courses: [courses.coreservlets.com](http://courses.coreservlets.com)

Slides © Marty Hall, <http://www.coreservlets.com>, book © Sun Microsystems Press